MAGAZINE

BSD

FOR NOVICE AND ADVANCED USERS

BSD in the Clouds

THE CLOUD IS AS SECURE AS YOU MAKE IT

PATTERNS FOR CLOUD INTEGRATION

THE CLOUD ITSELF IS NOT THE RISK... PATTERNS FOR CLOUD INTEGRATION

CLOUD SERVICE IN A DEVELOPER POINT OF VIEW

VOL.9 NO.10
ISSUE 74
1898-9144

FREENAS MINI
STORAGE APPLIANCE

IT *SAVES* YOUR LIFE.

HOW IMPORTANT IS YOUR DATA?

Years of family photos. Your entire music and movie collection. Office documents you've put hours of work into. Backups for every computer you own. We ask again, *how important is your data?*

NOW IMAGINE LOSING IT ALL

Losing one bit - that's all it takes. One single bit, and your file is gone.

The worst part? **You won't know until you absolutely need that file again.**

Example of one-bit corruption

THE SOLUTION

The FreeNAS Mini has emerged as the clear choice to save your digital life. **No other NAS in its class offers ECC (error correcting code) memory and ZFS bitrot protection to ensure data always reaches disk without corruption and *never degrades over time*.**

No other NAS combines the inherent data integrity and security of the ZFS filesystem with fast on-disk encryption. No other NAS provides comparable power and flexibility. The FreeNAS Mini is, hands-down, the best home and small office storage appliance you can buy on the market. **When it comes to saving your important data, there simply is no other solution.**

The Mini boasts these state-of-the-art features:

- 8-core 2.4GHz Intel® Atom™ processor
- Up to 16TB of storage capacity
- 16GB of ECC memory (with the option to upgrade to 32GB)
- 2 x 1 Gigabit network controllers
- Remote management port (IPMI)
- Tool-less design; hot swappable drive trays
- FreeNAS installed and configured

iXsystems™

http://www.iXsystems.com/mini

intel inside ATOM™

FREENAS CERTIFIED STORAGE

With over six million downloads, FreeNAS is undisputedly *the* most popular storage operating system in the world.

Sure, you could build your own FreeNAS system: research every hardware option, order all the parts, wait for everything to ship and arrive, vent at customer service because it *hasn't*, and finally build it yourself while hoping everything fits - only to install the software and discover that the system you spent *days* agonizing over **isn't even compatible**. Or...

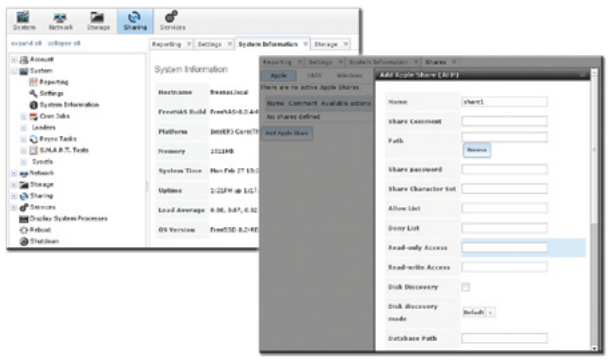
MAKE IT EASY ON YOURSELF

As the sponsors and lead developers of the FreeNAS project, iXsystems has combined over 20 years of hardware experience with our FreeNAS expertise to bring you FreeNAS Certified Storage. **We make it easy to enjoy all the benefits of FreeNAS without the headache of building, setting up, configuring, and supporting it yourself.** As one of the leaders in the storage industry, you know that you're getting the best combination of hardware designed for optimal performance with FreeNAS.

Every FreeNAS server we ship is...

- » Custom built and optimized for your use case
- » Installed, configured, tested, and guaranteed to work out of the box
- » Supported by the Silicon Valley team that designed and built it
- » Backed by a 3 years parts and labor limited warranty

As one of the leaders in the storage industry, you know that you're getting the best combination of hardware designed for optimal performance with FreeNAS. **Contact us today for a FREE Risk Elimination Consultation with one of our FreeNAS experts.** Remember, every purchase directly supports the FreeNAS project so we can continue adding features and improvements to the software for years to come. **And really - why would you buy a FreeNAS server from *anyone* else?**



FreeNAS 1U

- Intel® Xeon® Processor E3-1200v2 Family
- Up to 16TB of storage capacity
- 16GB ECC memory (upgradable to 32GB)
- 2 x 10/100/1000 Gigabit Ethernet controllers
- Redundant power supply

FreeNAS 2U

- 2x Intel® Xeon® Processors E5-2600v2 Family
- Up to 48TB of storage capacity
- 32GB ECC memory (upgradable to 128GB)
- 4 x 1GbE Network interface (Onboard) - (Upgradable to 2 x 10 Gigabit Interface)
- Redundant Power Supply

http://www.iXsystems.com/storage/freenas-certified-storage/

Dear Readers,

I'm proud to deliver a new issue of BSD Magazine to you. This time we are focused on Cloud computing. I hope that my words find you well and in a happy mood, as this is such an enjoyable topic. We hope you will read all the articles and we welcome any comments you may have.

We have collected the articles written by experts in their field to provide you with highest-quality knowledge. Enjoy your reading and develop your new skills with our magazine!

If you want to find out more about Attorney Confidentiality in Cloud Computing, check out the article provided by Benjamin Wright. Benjamin is an attorney in private practice (benjaminwright.us). He teaches Data Security and Investigations Law at the SANS Institute.

Also, we recommend that you read two short columns by Dan Srebick about Cloud Security: The Cloud is as Secure as You Make It and The Cloud Itself is Not the Risk. We hope you will enjoy them and let us know what you think about such short columns.

For my side, I would like to recommend that you read Cloud Service from a Developer Point of View by David Carlier. He is an experienced developer, is used to handling languages like C/C++, Java, Python, PHP, with Linux, *BSD and Win32 Operating Systems and has worked inside startups as well as bigger companies. He is a big fan of FreeBSD and C/C++ are his preferred programming languages most of the time.

Of course, please do not forget to read Mark VonFange's article: "FreeNAS: A Worst Practices Guide", and the amazing interview with Brian Callahan from the Devio us team! They try to create a tight-knit IT related community that's made up of geeks, developers, IT professionals and enthusiasts.

And for dessert, please go to see what Rob wrote for you this time. We really like his column and we are eagerly waiting to see what he will write next month.

As long as we have our precious readers, we have a purpose. We owe you a huge THANK YOU. We are grateful for every comment and opinion, either positive or negative. Every word from you lets us improve BSD magazine and brings us closer to the ideal shape of our publication.

Thank you. Marta & BSD Team

MAGAZINE BSD

Editor in Chief:
Marta Ziemianowicz
marta.ziemianowicz@software.com.pl

Contributing:
Michael Shirk, Andrey Vedikhin, Petr Topiarz, Solène Rapenne, Anton Borisov, Jeroen van Nieuwenhuizen, José B. Alós, Luke Marsden, Salih Khan, Arkadiusz Majewski, BEng, Toki Winter, Wesley Mouedine Assaby, Rob Somerville

Top Betatesters & Proofreaders:
Annie Zhang, Denise Ebery, Eric Geissinger, Luca Ferrari, Imad Soltani, Olaoluwa Omokanwaye, Radjis Mahangoe, Mani Kanth, Ben Milman, Mark VonFange

Special Thanks:
Annie Zhang
Denise Ebery

Art Director:
Ireneusz Pogroszewski

DTP:
Ireneusz Pogroszewski
ireneusz.pogroszewski@software.com.pl

Senior Consultant/Publisher:
Paweł Marciniak
pawel@software.com.pl

CEO:
Joanna Kretowicz
joanna.kretowicz@software.com.pl

Publisher:
Hakin9 Media SK
02-676 Warsaw, Poland
Postepu 17D
Poland
worldwide publishing
editors@bsdmag.org
www.bsdmag.org

Hakin9 Media SK is looking for partners from all over the world. If you are interested in cooperation with us, please contact us via e-mail: editors@bsdmag.org.

ServerU

Rack-mount networking server

Designed for BSD and Linux Systems

DESIGNED FOR FreeBSD | DESIGNED FOR GNU / Linux | DESIGNED FOR PRO apps FreeBSD Enterprise Appliance | DESIGNED FOR pfSense

Designed. Certified. Supported

Up to **5.5Gbit/s** routing power!

KEY FEATURES

- 6 NICs w/ Intel igb(4) driver w/ bypass
- Hand-picked server chipsets
- Netmap Ready (FreeBSD & pfSense)
- Up to 14 Gigabit expansion ports
- Up to 4x10GbE SFP+ expansion

PERFECT FOR

- BGP & OSPF routing
- Firewall & UTM Security Appliances
- Intrusion Detection & WAF
- CDN & Web Cache / Proxy
- E-mail Server & SMTP Filtering

contactus@serveru.us | www.serveru.us
8001 NW 64th St. Miami, LF 33166 | +1 (305) 421-9956

News

The FreeBSD Corner

Expert Says…

Security

Clouds Integration

Tips&Tricks

Unix

Interview

Column

Great Specials

On FreeBSD® & PC-BSD® Merchandise

Give us a call & ask about our
SOFTWARE BUNDLES
1.925.240.6652

$39.95
FreeBSD 9.1 Jewel Case CD Set
or FreeBSD 9.1 DVD

$29.95
PC-BSD 9.1 DVD

$49.95 Save a BUNDLE!
The PC-BSD 9.0 Users Handbook
PC-BSD 9.1 DVD

$99.95
The FreeBSD CD or DVD Bundle

Inside each CD/DVD Bundle, you'll find:
FreeBSD Handbook, 3rd Edition
Users Guide FreeBSD Handbook, 3rd Edition, Admin Guide
FreeBSD 9.1 CD or DVD set
FreeBSD Toolkit DVD

Stylish Dress Attire
Look Your Professional Best

Comfy Apparel
Stay Warm in Zip Ups & Pullovers

T-Shirts
Lots of Styles to Choose From

FreeBSD 9.1 Jewel Case CD/DVD $39.95

CD Set Contains:

- **Disc 1** Installation Boot LiveCD (i386)
- **Disc 2** Essential Packages Xorg (i386)
- **Disc 3** Essential Packages, GNOME2 (i386)
- **Disc 4** Essential Packages (i386)

FreeBSD 9.0 CD $39.95
FreeBSD 9.0 DVD $39.95

FreeBSD Subscriptions

Save time and $$$ by subscribing to regular updates of FreeBSD

FreeBSD Subscription, start with CD 9.1 ..... $29.95
FreeBSD Subscription, start with DVD 9.1 ..... $29.95
FreeBSD Subscription, start with CD 9.0 ..... $29.95
FreeBSD Subscription, start with DVD 9.0 ..... $29.95

PC-BSD 9.1 DVD (Isotope Edition)

PC-BSD 9.1 DVD $29.95
PC-BSD Subscription $19.95

The FreeBSD Handbook

The FreeBSD Handbook, Volume 1 (User Guide) $39.95
The FreeBSD Handbook, Volume 2 (Admin Guide) $39.95

The FreeBSD Handbook Specials

The FreeBSD Handbook, Volume 2 (Both Volumes) $59.95
The FreeBSD Handbook, Both Volumes & FreeBSD 9.1 ..... $79.95

PC-BSD 9.0 Users Handbook $24.95

BSD Magazine $11.99

The FreeBSD Toolkit DVD $39.95

FreeBSD Mousepad $10.00

FreeBSD & PCBSD Caps $20.00

BSD Daemon Horns $2.00

Bundle Specials!
Save $$$

Just Plain Fun
Mousepads & Novelty Horns

BSD Magazine
Available Monthly

For even **MORE** items
visit our website today!

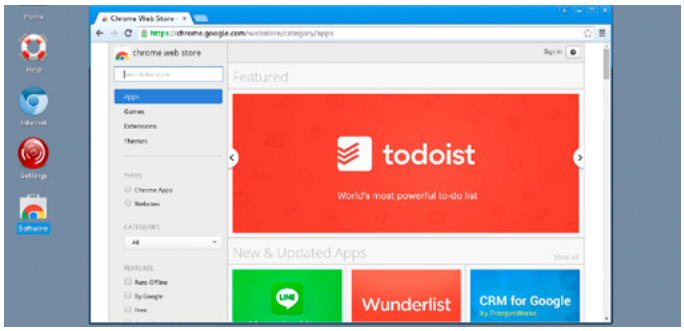
www.FreeBSDMall.com

DesktopBSD is releasing DesktopBSD 2.0 roadmap!

DesktopBSD is an open source operating system that aims to be a stable and powerful operating system for desktop users. Combining the stability of FreeBSD, the usability and functionality of KDE, and the simplicity of specially developed software to provide a system that's easy to use and install, in a project with two primary goals – security and usability.

Why DesktopBSD?

- Works out of the box, with full multimedia support and extremely easy to use.
- An operating system that respects your privacy, free of cost and open source for life.
- Users are encouraged to send feedback, their ideas will be heard.
- Provides a large software content ready to be installed from the Software manager.
- Modern, elegant and comfortable operating system which is both powerful and easy.



http://www.desktopbsd.net

KLEVV Urbane SSDs Released

KLEVV has entered the Solid State Drive market by releasing their first line of Urbane SSDs. Coming in three different storage capacities, 240GB, 480GB and 960GB, these 2.5-inch SSDs are equipped with an aluminum metal body, Toshiba 15nm MLC NAND Flash memory chips, a SATA 6.0 Gbps interface and a quad-core, 8-channel PHISON PS3110-S10 controller.

Designed for the fast life, the Urbane SSDs promise to deliver read/write speeds of up to 560/390 MB/s (240GB model) and 560/530 MB/s (480GB & 960GB models), respectively. In addition, users can also easily enjoy the service of KLEVV SSD Toolbox and Acronis True Image HD 2015 at the KLEVV Data Migration Software Center.

About the company

Established in 2014 with only 12 employees, Essencore started with one goal: to be the "Champion in Semiconductor distribution & Memory products." They have expanded the business domain worldwide and become an unprecedented Memory actor in the market.

The business strategies are to adopt the newest technologies to differentiate ourselves in front of customers from competitors, deliver dedicated memory products avoiding supply management issues, and offer various product portfolios for customer's competition readiness.

The company's core strengths are well-organized business structures, comprehensive product development and top-level human resources from the world's best Memory IDMs.

http://www.techfresh.net/klevv-urbane-ssds-released/

NextBSD recent updates

NextBSD is a code name for our “science project”, a name which is more tongue-in-cheek than serious. It started as an effort to adapt some of the more interesting Open Source technologies from Darwin/OS X to FreeBSD.

These technologies have collectively provided a higher level programming substrate for developers in OS X and iOS for many years now, replacing what have all too often been little more than semi-evolved shell scripts or bespoke solutions with limited architectural goals in other Unix variants.

NextBSD is also an effort to demonstrate that we need not be limited to simply one true FreeBSD. FreeBSD.org can and will continue to provide a conservative minimalist base for the development of advanced distributions like this one. Such distributions can make substantial additions to the basic core or reach different architectural decisions about which technologies to bundle in the core at all. And you may think of it as a “research laboratory” for such efforts, if that analogy helps.

In addition to the technologies that have received so much recent attention, we have also included VM optimizations from other vendors, as well as refinements to the network driver model. We are also eagerly seeking out other technologies that we believe merit inclusion, from new security technologies to fundamentally different approaches to packaging and distributing OS and appliance software.

- The basic ecosystem of launchd, notifyd, asld, and libdispatch work.
- These can be installed by cloning the NextBSD repo from github, building GENERIC or MACHTEST kernels, installing a new world on an existing 10.x or CURRENT system, and then following the instructions in the README.
- Launchd will start the initial jobs that are part of the repo now.

Google OnHub Router runs ChromiumOS (Chrome OS)

This is the same Linux-based operating system that powers Google Chromebook laptops and desktops.

OnHub is a modern dual-band wireless router, designed by Google and TP-Link, that operates networks on both the 2.4GHz & 5GHz frequency bands simultaneously and offers speed of up to 1900 Mbps.

Unlike traditional Broadband Routers, Google OnHub is designed to support “The Internet of Things” as well as other Smart devices, including Smartphones, Connected TVs and Computers.

A Team of Modders at Exploitee.rs, also famous as GTVHacker, have successfully managed to root a Google OnHub device in the same way they would with a Chromebook.

...And as an outcome of their reverse engineering on eMMC and the SPI flash dumps, the team discovered that the OnHub Router router runs something very similar to Google Chrome OS.

http://thehackernews.com/2015/10/root-google-onhub-chromeos.html

What's new in iOS

iOS 9 is Apple's newest operating system for iOS devices like the iPhone and the iPad, released to the public on September 16, 2015. iOS 9 builds on the content introduced with iOS 7 and iOS 8, bringing subtle design changes, refined features, improved functionality, and performance enhancements.

iOS 9's biggest focus is on intelligence and proactivity, allowing iOS devices to learn user habits and act on that information, opening up apps before we need them, making recommendations on places we might like, and guiding us through our daily lives to make sure we're where we need to be at the right time.

Siri is at the heart of the changes, and the personal assistant is now able to create contextual reminders and search through photos and videos in new ways. Swiping left from the home screen also brings up a new screen that houses "Siri Suggestions," putting favorite contacts and apps right at your fingertips, along with nearby restaurant and location information and important news.

The iPad's gotten some major feature additions in iOS 9, like split-screen multitasking that lets two apps be used at once and a picture-in-picture function that lets you watch a video while doing something else on the tablet. The keyboard on the iPad has deeper functionality with the addition of a new toolbar, and on both the iPhone and the iPad, there's a new two-finger swipe gesture that makes it easier to select content, cut, paste, and move the cursor on the screen.

Other changes include a new systemwide San Francisco font, wireless CarPlay support, an optional iCloud Drive app, built-in two factor authentication and optional longer passwords for better security.

Along with these features, iOS 9 features significant under-the-hood performance improvements. Battery optimizations provide an additional hour of battery use under typical conditions, and a new Low Power Mode further extends battery life up to three hours.

The current version of iOS 9 is iOS 9.0.2, which was released on September 30. iOS 9.0.2, like iOS 9.0.1, is a minor update that fixes several bugs. It fixes an issue that prevented app cellular data usage to be toggled on or off, resolves an issue that prevented iMessage activation, fixes an issue where an iCloud backup could be interrupted after starting a manual backup, and fixes a bug that could cause the screen to rotate incorrectly when receiving notifications. It also includes stability improvements for the Podcasts app.

Prior to iOS 9.0.2, Apple released iOS 9.0.1 on September 23. A minor update, iOS 9.0.1 introduced fixes for several bugs, including a glitch with the "Slide to Upgrade" screen that was preventing people from upgrading their devices from iOS 8 to iOS 9.

Apple is also testing the first major update to iOS 9, iOS 9.1. iOS 9.1 introduces features for upcoming products like the Apple TV, and it includes new emoji like unicorn head, cheese wedge, taco, middle finger, burrito, popcorn, and more. Thus far, Apple has seeded five betas of iOS 9 to developers and public beta testers.

http://www.apple.com/ios/whats-new/
http://www.macrumors.com/roundup/ios-9/

BBC bypasses Linux kernel to make streaming videos flow

It's no surprise, then, to learn of other high-performance efforts addressing the same issue: both the BBC in its video streaming farms; and CloudFlare, which needs to deal with frequent packet flood attacks.

High-definition video streams have to push out 340,000 packets per second into 4 Gbps ultra-high definition streams. With just 3 µs per packet of processing time, using the kernel stack simply wasn't an option.

Using the network sockets API, the post explains, involves a lot of handling of the packet, as "each data packet passes through several layers of software inside the operating system, as the packet's route on the network is determined and the network headers are generated. Along the way, the data is copied from the application's buffers to the socket buffer, and then from the socket buffer to the device driver's buffers."

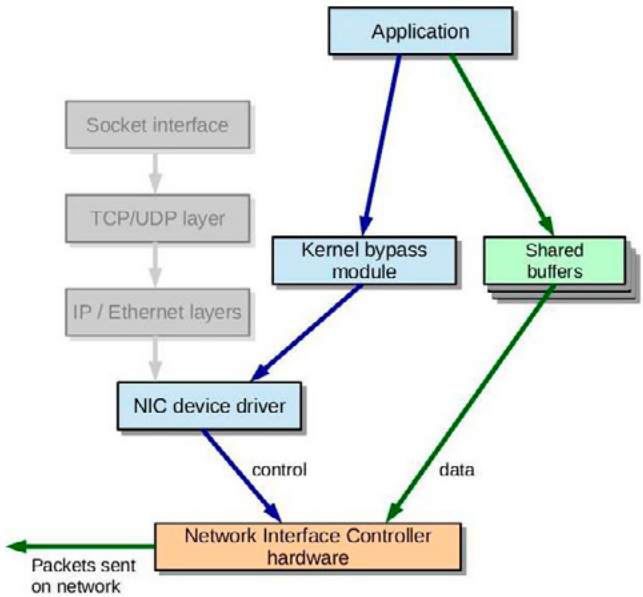
The boffins started by getting out of the kernel and into userspace, which let them write what they call a "zero-copy kernel bypass interface, where the application and the network hardware device driver share a common set of memory buffers".

The application creates a group of packets and their network headers, it does so directly in those shared buffers.

"Then using a single function call, the whole group is handed over to the control of the device driver which transmits them directly on to the network".

CloudFlare's approach is similar – a userspace kernel bypass – but with wrinkles specific to its circumstances.

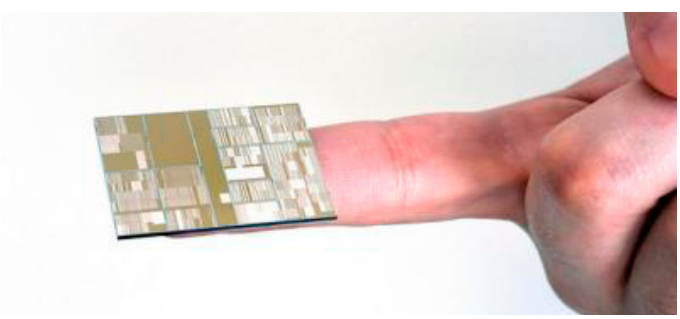
CloudFlare's problem is not just the quantity of packets, but the need to distinguish attack packets from user data. Regular readers of The Register will already know that the provider suffers regular attacks.



http://www.theregister.co.uk/2015/10/12/linux_networking_api_showing_its_age/

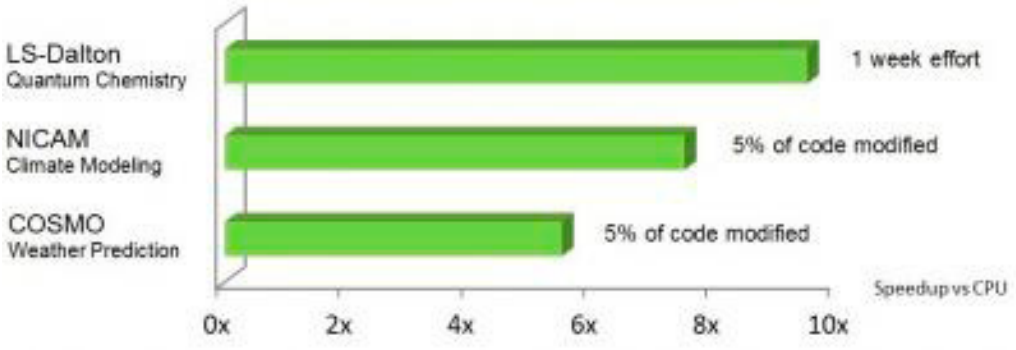
IBM Research Alliance's 7nm Node Chips

The secret to packing a whopping 20 billion transistors onto a fingernail-sized chip involves a combination of Silicon Germanium (SiGe) channel transistors and Extreme Ultraviolet (EUV) lithography integration. This formula, championed by an alliance led by IBM Research, is billed as the semiconductor industry's first 7nm node chip with functioning transistors. Today, microprocessors leverage 22nm and 14nm technologies, and 10nm is on its way to maturity. The new 7nm technology in the IBM consortium's test chips is considered critical to meeting the anticipated demands of future cloud computing and Big Data systems, cognitive computing, mobile products and other emerging technologies. Other partners in the public-private consortium include GLOBALFOUNDRIES, Samsung and the SUNY Polytechnic Institute's Colleges of Nanoscale Science and Engineering.



NVIDIA OpenACC Toolkit

It is not that computing cores aren't getting faster. Instead, processors are getting more parallel, which is a trend that is likely to continue. To harness advances in parallel computing, NVIDIA and its partners developed the OpenACC standard, which NVIDIA says "simplifies parallel programming for modern processors, like GPUs". In order to simplify access to OpenACC for researchers, NVIDIA has released the new NVIDIA OpenACC Toolkit, a free, all-in-one suite of OpenACC parallel programming tools. NVIDIA claims that scientists can do "more science, less programming" from the solution, which features "the industry-leading" PGI Accelerator Fortran/C Workstation Compiler Suite for Linux. The compiler is free to academic developers and researchers. The toolkit also includes the NVProf Profiler, which gives guidance on where to add OpenACC "directives"—that is, simple compiler hints to accelerate code, as well as simple, real-world code samples.



The Qt Company's Qt

The motto for the Qt Company is simple: "Code less. Create more. Deploy everywhere." It's a sensible leitmotif given that the company made the new Qt 5.5, the upgraded C++-based framework of libraries and tools for developing powerful, interactive and cross-platform applications and devices. Qt's support for multiple desktop, embedded and mobile operating systems allows developers to save significant time on application and device development simply by reusing one code. The most notable innovations in Qt 5.5 are the following: full Bluetooth Low Energy for Internet of Things deployments, a pre-built version of Qt for RHEL 6.6 and preliminary support for upcoming Windows 10 (full subsequent support to follow with a patch release). Other new features include extended support for multimedia and graphics creation with 3D capabilities, as well as new multi-screen and IoT development features that strengthen overall performance across applications and devices.

http://www.linuxjournal.com/slideshow/new-products-8

Among clouds Performance and Reliability is critical

Download syslog-ng Premium Edition product evaluation here

Attend to a free logging tech webinar here

BalaBit IT Security

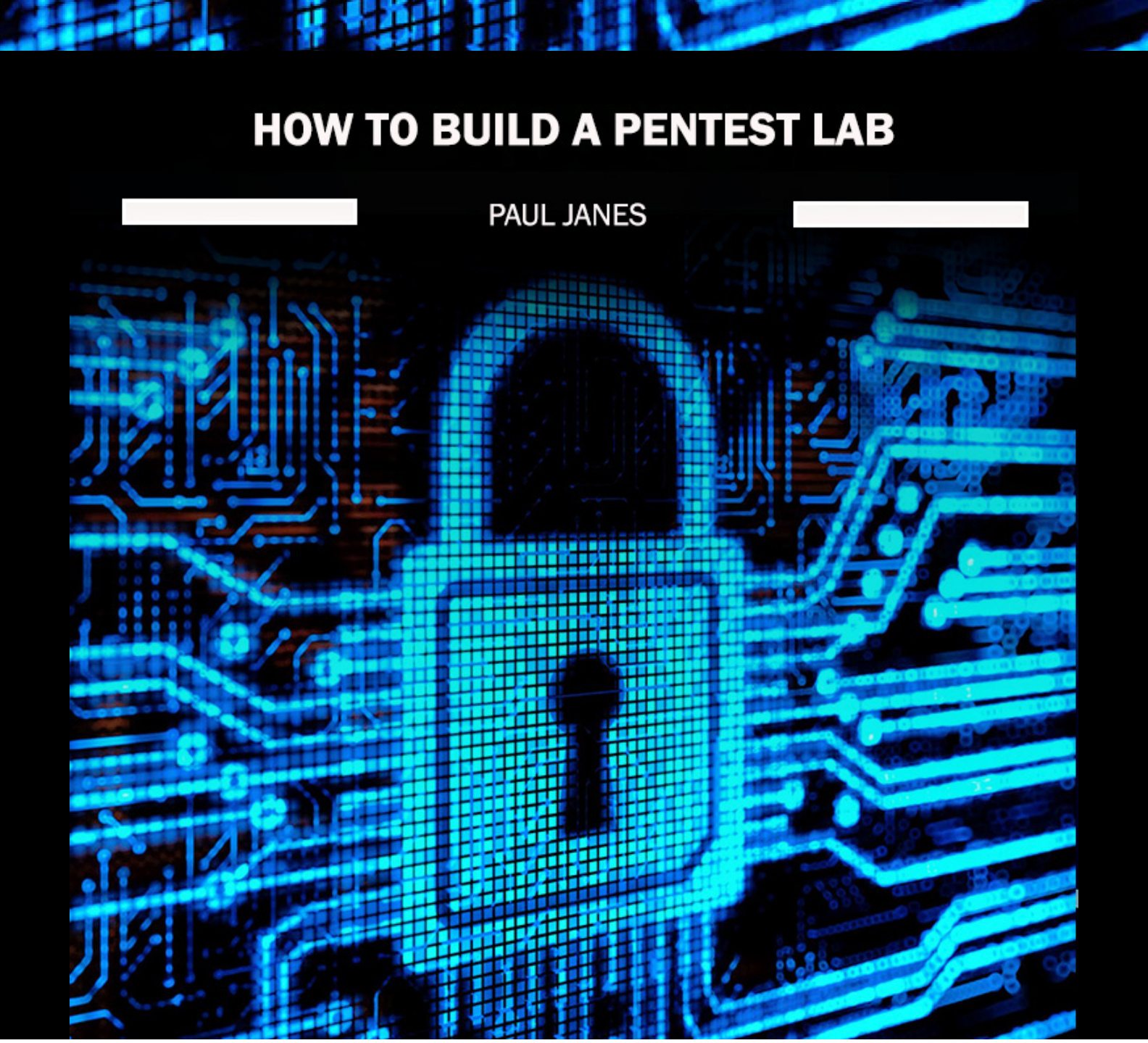
www.balabit.com

syslog-ng log server

The world's first High-Speed Reliable Logging™ technology

HIGH-SPEED RELIABLE LOGGING

- above 500 000 messages per second
- zero message loss due to the Reliable Log Transfer Protocol™
- trusted log transfer and storage



Enroll to BUILD YOUR OWN PENTEST LAB online course and learn how to create your own pentest lab.

This course covers various virtualization software and penetration testing tools like Kali Linux, Nessus, Metasploit, Metasploitable, Nmap, and others.

Through practical hands-on labs, you will be able to not only identify systems but also identify their vulnerabilities.

All in pure practice.

In case of any questions please contact:

joanna.kretowicz@eforensicsmag.com

Course Plan:

Pre-Course Material

- Why Do I Need a Pen Test Lab
- Definitions
- Creating Directory Structure For the Course
- Download Virtual Images
- Acquire Nessus Licenses

Module 1 The Build

- Definitions
- Some Basic Linux Commands You Need to Know

Software

- Installation of VMPlayer and Virtual Box. You Decide, We Will Cover Both.
- Setup of Our Penetration Testing System – Kali Linux Distribution
- Setup a Linux Client as a Virtual Machine
- Setup Our First Vulnerable Machine Metasploitable2
- Setup Our Second Vulnerable Machine Bee-box (BWAMP)

Exercises

- Overview of Virtual Machine Settings
- Run the Basic Linux commands
- Upgrade Kali Linux Distribution

Module 2 Port Scanning

- Nmap and Zenmap Installation
- Nmap Basic Scanning
- ZenMap Basic Scanning
- Metasploitable Dnmap Scanning

Exercises

- Run Nmap Scans against Ubuntu
- Run Zenmap Scans Against Metasploitable2
- Run Dnmap Scans Against Host

Module 3 Vulnerability Scans

- Installation and Licensing of Nessus Vulnerability Scanner
- Installation of Netsparker Web Vulnerability Scanner
- Basic Nessus Scanning
- Basic Netsparker Scanning
- Intermediate Nmap Scans

Exercises

- Run a Nessus Scan Against Metasploitable2
- Run a Netsparker Scans Against Bee-Box (BWAMP)
- Run a Nessus Scan Against Ubuntu

Module 4 Advanced Scanning and Reporting

- Nessus Advanced Scans
- Netsparker Advanced Scans
- Nmap Advanced Scans
- Metasploit Reporting
- Review Other Resources Available to You…
- Where Do I Get Virtual Machines

Exercises

- Create a Metasploit Report Combining Nessus and Dnmap Scans
- Run an Advanced Nessus Scan Against Metasploitable 2
- Run an Advanced Netsparker Scan Against Bee-Box (BWAMP)

If you have any questions or just want to get to know us better feel free to contact me at joanna.k@eforensicsmag.com or just answer this email

Cloud Computing – BSD in the CLOUDS

OLAOLUWA OMOKANWAIYE

There is a saying in my culture – "the clouds are wide enough for every bird to fly without hindering another". The meaning of this profound proverb is that there are more than enough opportunities for everyone without any problems whatsoever, and in the case of our discourse about the Cloud, it does play.

It is the greatest time for open source professionals and the BSD Community. This "Cloud Technology movement" in computing world is already obvious. Cloud vendors are experiencing growth rates of 50% per annum as more users are demanding cloud services. RightScale conducted its fourth annual State of the Cloud Survey of the latest cloud computing trends at the beginning of this year 2015, with a focus on infrastructure-as-a-service. From the survey of some 930 IT professionals asked, it showed that there's a greater adoption of cloud infrastructure and related technologies. The respondents included technical executives, managers and practitioners and represented organizations of varying sizes across many industries. Again this showed that more and more companies are moving into cloud use for one service or the other. Even tech giants are providing and making more of their services available in the cloud. Moreover a large percentage of the applications and technologies used by individuals always have a cloud feature to work online or even provide back-ups, think of your favorite online application like a word-processor such as Google-docs, or file-sharing application like Drop Box and Google Drive, or the tool you use for organising like Keep or Evernote and even your favorite photo app. In fact, most developers see it as a must and an added advantage adding a cloud feature to their products.

Just as the proverb I began this article with, presently, there are more than enough opportunities for the entire BSD community, BSD professionals, as well as BSD service start-ups concerning cloud technologies and here's why:

I predict that the following statements are, or soon will be, true:

- The next billion dollar business is in the cloud;
- More companies are firing up BSD in their data center and clouds;
- BSD-savvy professionals are in high demand;

Let's take each of these points one at a time to discuss.

The next billion dollar business is in the cloud

The article "Here's Where Amazon and Google Could Make Their Next $100B" caught my eye a few days ago showcasing a report put together by the tech industry research outfit, Forrester, predicting the future of the cloud computing business. Forrester's report, drew from interviews with vendors and customers across the market, focused mainly on "public cloud services" – Internet services, like those from Amazon and Google and Microsoft, that allow businesses build and operate software without setting up their own hardware. The new report predicts that this market will grow to $191 billion by 2020. That's 20 percent more than they predicted in their previous report, back in 2011. "The adoption among cloud among enterprises, which is really where the money is, has really

picked up steam," John Rymer, vice president at Forrester, said, "It's a big shift. The cloud has arrived. It's inevitable", he further added.

Rymer and Forrester identified the cloud, especially the public cloud (offering cloud to all), as being a "hyper-growth" market. The report further shows that, this "hyper-growth" market made up of "cloud platform services" like Amazon EC2, will be a $44 billion market by 2020, back-end business services will reach $14 billion, and cloud software applications will hit $131 billion. "A lot of businesses are now saying: 'I want to move my operational application, back office applications, into public clouds,'" John Rymer said, he also said, "...in the past, so many people said: 'I'm never going there. Now they're actually working at it.". And as though that was not enough Bloomberg Business recently published an article, "Cloud Boom Boosts Google, Amazon With $90 Billion Stock Surge", revealing the success of the cloud shift everyone was talking about, and how Google and Microsoft and Amazon are already benefiting from as evidenced in their stocks, all hitting record high. For example, Amazon Web Services division, soared from 78 percent from a year ago with sales of $2.09 billion. As of this writing, Amazon's stock gained as much as 10 percent to $619.45, Microsoft added 11 percent to $53.16, and Google soared 12 percent to $730, as revealed by Bloomberg.

Google's CEO, Sundar Pichai already said, "Every business in the world is going to run on cloud eventually."

More companies are firing up BSD servers in the cloud, for cloud services and even more will start

From the well known names like Digital Ocean, Open Stack, Google and Amazon to others like CloudSigma and BSDvm, more and more companies are serving the latest BSD versions (especially the FreeBSD 10) in the cloud, both for customers and developers.

For example, the back end of WhatsApp, a mobile services platform acquired by Facebook in October 2014 at a final price that topped $21.8 billion runs on FreeBSD 10. FreeBSD appeals more to some developers for the back end of heavily trafficked systems, given its reputation as a stable, 30-year old version of Unix. Offering FreeBSD, has made DigitalOcean, for example, ahead of the major cloud suppliers when it comes to appealing to developers.

The customers desire and demand to have their favourite OS in the cloud as well, to enjoy the many benefits and advantages that BSD offers such as the robust community, the OS stability, security, ease of use, the many ports available among many other benefits.

And guess what? The reviews about the BSD services are just splendid. One person said in one review, "we can all now enjoy FreeBSD on AWS..., this was a long time coming. And finally wait is over", another said, "Works flawlessly. Deployed in a minutes", while yet another said, "I got my FreeBSD instance up and running with just a few clicks, … performance is great".

BSD-savvy professionals are in high demand

To fill up these data centres and help operate the cloud servers you need capable people. Take for example, the Linux Professional Institute (*https://www.lpi.org*) certifies Linux Professionals and the BSD Certification Group (*http://www.bsdcertification.org/*) certifies BSD professionals. Both organisations are non-profit organisations committed to creating and maintaining global certification standards for system administration on Linux and BSD based operating systems respectively and help candidates gain the necessary skills.

The LPI Certification has the Linux Essentials Professional Development Certificate, Linux Server Professional Certification (LPIC-1), Linux Network Professional Certification (LPIC-2), Linux Enterprise Professional Certification (LPIC-3).

The BSD Certification Group has two levels of certification – the BSD Associate (BSDA), an entry level certification on BSD systems administration and the BSD Professional (BSDP) designed to be an advanced certification for senior system administrators with at least three years of experience on BSD systems. These exams are thorough and based on psychometric making sure they reflect the needs of the IT community and industry. Once a candidate is well prepared either by professional training and self study accompanied with lab practise, the exams are nothing to be afraid of. Also the objectives and list of study materials can be found on their websites.

The value of these certifications cannot be over-emphasized as more organisations are requiring proof of professionalism from applicants, employees and consultants.

Indeed.com is an excellent job board where Linux/UNIX/BSD professionals can find job vacancies (from network Engineers to System Administrators, Security Specialists, support technicians and many more) – these job offers always request for *NIX/BSD skills either specifically or as an added advantage. Applicants are also encouraged to upload their resumes ahead, so employers can find them easily. Now imagine what happens when an employer sees in your resume that you are certified by a standard body like LPI or BSDCG coupled with the experience you have in the field, at that point your certification speaks for you.

LinkedIn is another place, one of the best business platforms where BSD and Open Source Pros can find jobs or to find companies interested in their skills.

At this point, the *NIX/BSD experts need to be heavily involved in cloud technologies, and market themselves, (thankfully there's social media). If you think it is not that obvious that professionals with such skills are a necessity, just read Google Cloud Platform web page: Google says that, for the operating system images in the table listing given, (which incidentally included FreeBSD10) that support for the OSes including BSD can be gotten with the resource listed under Support channel. It further says that, "Compute Engine does not manage these operating systems and any questions or costs would be determined by the corresponding operating system community."

First, I think, it is good that a tech giant, like Google (that is gaining increased ground in the cloud industry), provides support for the BSD OS. Even though its compute engine does not manage BSD OS, it emphasizes by saying (and here's the catch and opportunity for the BSD community and experts) – "...any questions or costs would be determined by the corresponding OS community" (for BSD, that would be the BSD community). This shows that the cost and advise to provide these services are determined and controlled by the BSD savvy professional. Now that is a great opportunity. For instance, BSD savvy experts can offer the services required in this regard, either by building start-ups, or by forming partnerships with these cloud-offering organisations or even by being part-time or full-time employees to the cloud organisations to fill in this gap and also make some money.

As we know, Amazon is another giant in the cloud business. This further confirms the significant space the BSD OS holds and the community members (like Colin Percival, a FreeBSD contributor and FreeBSD security officer, whose name came up in the reviews) involved in bringing the BSD cloud instance live for customers on the AWS infrastructure. As mentioned earlier, customer reviews about the FreeBSD instance in the AWS cloud space are terrific. Amazon is one of the big players and is growing. Having the required BSD skills to deploy and manage such in the AWS cloud platform is a big advantage, giving a BSD professional an edge over others as it is readily demanded but often scarce.

Aspiring BSD professionals can start developing themselves now, resources are readily available online – tutorials, training resources, forums, online programs and a whole community available to assist. BSD Conferences are also held consistently around the globe. Diligent practice is a must on the part of the aspiring BSD pro. You can work with virtualization software or work directly on an available system either personal or in the cloud, now easy and affordable. Certification programs are sure advantages.

Just as resources are vast, so are the opportunities, lets be a part of it and seize the opportunity this moment.

God bless the BSD and Open Source Software community.

God bless you.

This article was written by Olaoluwa Omokanwaiye.

Olaoluwa is a Linux professional and a BSD user and BSD magazine beta-tester. He works with the Linux Professional Institute Master Affiliate in Nigeria.

He started using the *NIX OSes many years back when he got inquisitive and curious with operating systems other than the popular proprietary ones. He is happily married to Eniola, an architect and interior designer, and they have an adorable daughter, Grace-Lois that is already tinkering with her dad's Android tablet.

Footnotes

- BSD Cert Group: *http://www.bsdcertification.org/*
- LPI means Linux Professional Institute : *www.lpi.org*
- See details on RightScale's Cloud Survey in January 2015 here: *http://www.rightscale.com/blog/cloud-industry-insights/cloud-computing-trends-2015-state-cloud-survey*
- See details on why cloud vendors are growing 50% per annum here: *https://en.wikipedia.org/wiki/Cloud_computing#cite_note-12*
- For more details on the article "Here's Where Amazon and Google Could Make Their Next $100B", see *http://www.wired.com/2015/10/amazon-google-make-next-100-billion/*
- For more details on Bloomberg Business article, "Cloud Boom Boosts Google, Amazon With $90 Billion Stock Surge", see *http://www.bloomberg.com/news/articles/2015-10-23/the-cloud-is-raining-cash-on-amazon-google-and-microsoft*
- See FreeBSD now gaining ground with small cloud providers here: *http://www.informationweek.com/cloud/infrastructure-as-a-service/freebsd-gains-ground-with-small-cloud-providers/d/d-id/1318656*
- For benefits of FreeBSD in cloud and as VPS see *https://www.atlantic.net/blog/freebsd-ssd-cloud-vps-hosting-10-reasons/*
- See Operating systems with support out of compute engine on: *https://cloud.google.com/compute/docs/operating-systems/#operating_systems_with_support_outside_of_short_product_name*
- For more details on Amazon Reviews about FreeBSD instance see: *https://aws.amazon.com/marketplace/review/product-reviews/ref=dtl_pop_customer_reviews/182-4702837-2618265?ie=UTF8&asin=B00KSS55FY*

ABOUT THE AUTHOR

Olaoluwa Omokanwaiye ,has been passionate about open source technologies since his sophomore year. Today, twelve years later, he has worked with the Linux Professional Institute since 2008 in promoting and providing training and certification opportunities to a wide range of clients from server administrators to air traffic control personnel. A keen advocate of open source systems, he is presently working with a team that's looking to set up an innovation hub at a prestigious university in Nigeria. He is married to Eniola, an architect, and their 1 year old daughter is fast becoming a pro at tinkering with her Dad's Android tablet. In his free time Laolu watches Marvel movies, follows up on new developments in robotics and plucks a few strings on his violin. Connect with Olaoluwa on LinkedIn at ng.linkedin.com/in/olaoluwa twitter at dnachild.

NET OPEN SERVICES IS AN APPLICATION HOSTING COMPANY FOCUSED ON OPEN SOURCE APPLICATIONS MANAGEMENT IN HIGH AVAILABILITY ENVIRONMENT.

NET OPEN SERVICES IS PROUD TO PROVIDE A HIGH QUALITY SERVICE TO OUR CUSTOMERS SINCE 10 YEARS.

OUR EXPERTISE INCLUDES:

- **CLOUD COMPUTING, PUBLIC, PRIVATE AND HYBRID CLOUD MANAGEMENT (OPENSTACK, CLOUDSTACK, RED HAT ENTERPRISE VIRTUALIZATION)**
- **REMOTE MONITORING AND MANAGEMENT 24/7**
- **NETWORKING AND SECURITY (OPEN BSD, IP TABLE, CHECKPOINT, CISCO,...)**
- **OS AND APPLICATION MANAGEMENT (FREE BSD, OPEN BSD, SOLARIS, UNIX, LINUX, AIX, MS WINDOWS)**
- **DATABASE MANAGEMENT (ORACLE, MYSQL, CASSANDRA, NOSQL, MS SQL, SYBASE...)**
- **MANAGED HOSTING IN CARRIER CLASS DATA CENTERS**
- **DISASTER RECOVERY**

Net Open
SERVICES

WE PROVIDE SERVICES IN EVERY STEP OF THE PROJECT LIFE, DESIGN, DEPLOYMENT, MANAGEMENT AND EVOLUTIONS. **NETOPENSERVICES** TEAM INCLUDES EXPERIENCED LEADERS AND ENGINEERS IN THE INTERNET SERVER INDUSTRY.

OUR TEAM HAS **15 YEARS OF EXPERIENCE** IN DEVELOPING INTERNET INFRASTRUCTURE-GRADE SOLUTIONS AND PROVISIONING INTERNET DATACENTERS AND GLOBAL SERVICE NETWORKS TOGETHER.

WE OFFER EXCEPTIONAL HARDWARE SUPPORT AS SOFTWARE SUPPORT ON UNIX/LINUX AND OPEN SOURCE APPLICATION. **NETOPENSERVICES** DELIVERS THESE CUSTOM-BUILT LINUX AND UNIX SERVERS, AS WELL AS PRECONFIGURED SERVERS AND SCALABLE STORAGE SOLUTIONS, TO OUR CUSTOMERS. WE ALSO OFFER CUSTOM DEVELOPMENT AND ADVANCED-LEVEL UNIX/LINUX CONSULTING SOLUTIONS.

WWW.NETOPENSERVICES.COM • CONTACT@NETOPENSERVICES.COM

Attorney Confidentiality in Cloud Computing No Trespassing Banners May Be Effective

BY BENJAMIN WRIGHT

Are attorney records stored in the cloud accorded confidentiality by law?

I don't have the final answer to that question, but I do have some ideas to promote confidentiality.

The confidentiality of attorney records is normally based on two legal doctrines – attorney-client privilege and attorney work product.

Evidence That Maybe Attorney Records Are Not Being Accorded Confidentiality

Five recent developments raise questions about the confidentiality of the digital records belonging to lawyers.

- **Item One:** According to rumor, national intelligence agencies have tapped into law firm records and communications. Allegedly, a document leaked by Edward Snowden shows that the Australian Signals Directorate, in cooperation with the US National Security Agency, spied on a US law firm (rumored to be Mayer Brown) that was advising the government of Indonesia in trade negotiations. Allegedly, the government received legal advice in support of its spying on the firm.*
- **Item Two:** The FBI has informed some US law firms that they have been hacked by bad guys. Some have speculated that the reason the US government possesses this knowledge is that the US government itself was also spying on the law firms.*
- **Item Three:** A whiff of uncertainty has emerged about whether lawyers are wise to store records in the cloud. One school of thought argues that the cloud provider is a third party (that is, not the lawyer and not the client). This school argues that by placing the records in the hands of the third party, and arguably allowing the third party to monitor the records in some way, the lawyer has waived confidentiality rights.
- **Item Four:** Microsoft – the cloud service provider for Hotmail (a.k.a. Outlook.com) – surreptitiously searched the contents of a Hotmail account belonging to an independent blogger who did not work for Microsoft. Microsoft did not see prior approval from a court or other government authority. Microsoft believed its action as service provider was justified by evidence that the blogger's Hotmail account was connected with infringement of Microsoft's intellectual property.
- **Item Five:** British spies believe they have legal authority to inspect confidential lawyer records and communications.

Human Rights

Can Banners Effectively Increase Confidentiality?

Given these presumably disturbing developments, is there anything lawyers can do? I propose lawyers mark their records with banners and notices of confidentiality.

It is inexpensive to post legal banners and notices to assert zones of confidentiality. Although there is no guarantee that law will respect banners and notices, there is no guarantee that it will not respect them. So I publicly publish the following declaration on my OneDrive page. (OneDrive is a Microsoft cloud computing service for storing files.)

Publish This Claim With Cloud-Stored Records

NO TRESPASSING. ALL FILES STORED ON BENJAMIN WRIGHT'S ONEDRIVE ACCOUNT ARE PRIVATE, PROPRIETARY AND CONFIDENTIAL UNLESS THEY ARE CONFIGURED BY MR. WRIGHT TO BE ACCESSIBLE TO THE PUBLIC.

Benjamin Wright is licensed as an attorney. Some of Mr. Wright's non-public records stored in the cloud are subject to confidentiality protections associated with attorney work and communications. The laws of many countries recognize such protections. Wright insists that you recognize those protections with respect to his records and communication.

Video Version May Carry More Rhetorical Weight

On my OneDrive account I publicly publish a video version of the same claim. (*https://www.youtube.com/watch?t=5&v=dgjFFQgZcus*).

Post this Notice at Data Center

What could the owner of a cloud or hosting service do to bolster the legal protections afforded to lawyer or client data stored in the service? One idea is to post a legal notice.

Below is a notice that could be posted physically at the service's data center and on administrative log-on screens connecting to the center. One of the goals of this notice is to persuade any American authority that it should, under American law and policy, respect the property and privacy rights associated with the data. This effort in persuasion might apply, for example, to:

- a court-issued subpoena
- a duly-authorized tax summons
- a physical police raid
- a surreptitious online government break-in

This data center hosts data that is the property of other organizations. Most of this data is sensitive. Much of it is protected by privileges associated with attorney work on behalf of clients. Much of it relates to private, personally-identifiable information about individuals. The laws of United States and the laws of many other countries respect rights and privileges related to property, attorney work and individual privacy.

The United States observes the rule of law. As evidenced by the US Constitution and many other American laws, privacy is a fundamental human right in the United States.

Mismanagement of the data in this data center can cause great damage. Anyone – including a government official – tampering with or hindering the lawful use of this data is advised to act with care and diligence.

Anyone who, by legal authority, seeks to access or impede data in this center is advised that through the use of skill and diligence, his or her lawful mission can be accomplished without infringing the rights of bystanders, such as non-involved customers and individuals.

A law firm might post similar notices on its internal computers.

Dear reader: what do you think about this topic?

***Footnote:** I don't know beans about what national intelligence agencies do or don't do. I am not passing judgment on any particular event. But modern developments in technology and surveillance do justify a larger discussion of confidentiality law.

Postscript: The form legal language I publish above is not copyrighted. It is just form legal boilerplate based on stock legal verbiage. It is worthy of public use and discussion. Anyone may use it. But if you need legal advice or services, you should hire a lawyer.

ABOUT THE AUTHOR

Benjamin Wright is an attorney in private practice. benjaminwright.us He teaches the Law of Data Security and Investigations at the SANS Institute.
https://www.sans.org/course/law-data-security-investigations

FreeNAS: A Worst Practices Guide

BY MARK VONFANGE

FreeNAS®

There are many best practices guides for managing storage solutions out there, but a lot of how you administer your storage depends on your specific use case and what you're trying to accomplish. While we have created a best practices for FreeNAS, we also decided to take a look at what you don't want to do; things that will leave you hurting either immediately or down the road.

In that spirit, we've put together a worst practices guide for FreeNAS based on years of experience with systems in the field. The easiest way to avoid these pitfalls is to simply purchase a TrueNAS system from the experts at iXsystems, who can help set up your systems for optimal performance and functionality. For those who prefer the DIY approach, here are some things to look out for when setting up and managing your own FreeNAS system.

Using Hardware RAID with ZFS

When setting up a RAID array, common knowledge says that hardware RAID is preferable to software RAID. This is something of a misconception as all RAID is software RAID. If you're using a hardware RAID controller, it has its own independent operating system that communicates with your disks and often has caches to improve read and write performance. This was a good idea in the distant past, and improved RAID performance substantially, but operating systems and the hardware they run on have come a long way since those days.

FreeNAS uses the ZFS file system and is designed to communicate directly with your disks using its own volume manager. ZFS includes a sophisticated yet efficient strategy for providing various levels of data redundancy, including the mirroring of disk and the "ZFS" equivalents of hardware RAID 5 and higher with the ability of losing up to three disks in an array. If a given set of disks is provided to ZFS using a hardware RAID card, ZFS will not be able to efficiently balance its reads and writes between them or rebuild only the data used by any given disk. Hardware RAID cards typically rebuild disks in a linear manner from beginning to end without any regard for their actual contents.

The "one big disk" that hardware RAID cards provide limits some of ZFS's advantages, and the read and write caches found on many hardware RAID cards are how risk gets introduced. ZFS works carefully to guarantee that every write it receives from the operating system is on disk and checksummed before reporting success. This strategy relies on each disk reporting that data has been successfully written, but if the data is written to a hardware cache on the RAID card, ZFS is constantly misinformed of write success. This can work fine for some time but in the case of a power outage, catastrophic damage can be done to the ZFS "pool" if key metadata was lost in transit. Such failures have been known to carry five-figure price tags for data recovery services. Unlike hardware RAID, you will not suffer from data loss that can occur from interrupted writes or corrupt data returned from a hardware cache with ZFS.

Finally, most hardware RAID cards will mask the S.M.A.R.T. disk health status information that each disk provides. Very simply, each disk is connected to the hardware RAID controller card and the disks become invisible to the standard S.M.A.R.T. monitoring utility "smartctl". Without access to this information, the user is left unaware

of classic warning signs of impending disk failure, like reallocated sector count or unusually high temperature. Even the time it takes to run smartctl can be indicative of an impending problem.

While some hardware RAID cards may have a "pass-through" or "JBOD" mode that simply presents each disk to ZFS, the combination of the potential masking of S.M.A.R.T. information, high controller cost, and anecdotal evidence that any RAID mode is about 5% slower than non-RAID "target" mode results in zero reasons for using a hardware RAID card with ZFS.

Long story short, using hardware RAID on FreeNAS can lead to anything from corrupted writes to fatal errors that require you to invest in costly data recovery services.

Setting up Deduplication without Adequate Planning

Deduplication is a much-desired feature for storage solutions. On any given system, more than half your data may be duplicates of data elsewhere in your storage pool, causing a greater storage consumption. Deduplication reduces capacity requirements significantly and improves performance by tracking duplicate data with a 'deduplication table', eliminating the need to write and store duplicate information. ZFS stores this table on disk, which means that, if the host has to refer to the on-disk tables regularly, performance will be substantially reduced because of the slower speeds of standard spinning disks.

This means you need to plan to fit your entire deduplication table in memory to avoid major performance and, potentially, data loss. This generally isn't a problem when first setting up deduplication, but as the table grows over time, you may unexpectedly find its size exceeds memory. This splits the deduplication table between memory and hard disk, turning every write into multiple reads & writes, slowing your performance down to a crawl. In an enterprise environment, this can cause significant productivity decreases and angry staff workers. If this happens, the best solution is to add more system memory so that the pool will be able to import back to memory. Unfortunately, this can sometime take days to perform, and, if your hardware already has maxed out its memory capabilities, would require migrating the disks to a whole new system to access the data.

The general rule of thumb here is to have 5 GB of memory for every 1TB of deduplicated data. That said, there may be instances where more is required, but you will need to plan to meet the maximum potential memory requirements to avoid problems down the road. To get a more precise estimate of the required memory for deduplication, do the following: run the 'zdb -b (pool name)' command for the desired pool to get an idea of the number of blocks required, then multiply the 'bp count' by 320 bytes to get your required memory. If it's less than 5GB, still use the 5GB per terabyte of storage rule. If it's higher, go with that number per terabyte.

For must use cases, it is recommended to just utilize lz4 compression for data consumption savings, as there's no real processing cost. In fact, due to of the advances in CPU speeds, compression actually improves disk performance because writing uncompressed data to disk takes longer than compressed data. To be safe, always use compression instead of deduplication unless you know exactly what you are doing.

Striping Without Redundancy

ZFS offers all the typical forms of RAID redundancy and more, including ZFS striping (RAID 0), ZFS mirroring (RAID 1), RAID 10, and RAID-Z levels that allow for 1, 2 or 3 disk failures without affecting your storage pool. ZFS striping can speed up your performance by spreading out writes across multiple disks and combining all your disks into one large pool. *This can seem appealing to the new user because of its maximum speed and capacity, but if any of your disks has a failure, your entire pool will be lost.* While, with secondary storage or non-critical data, this may not prove to be a catastrophic loss, losing your storage pool is always a big deal and it's always recommended to configure your storage pool with some level of redundancy.

Using a SLOG for asynchronous write scenarios

The ZFS filesystem can tier cached data to help achieve sizable performance increases over spinning disks. Users can set up flash-based L2ARC read cache and SLOG (Separate ZFS Intent Log, sometimes called a ZIL) 'write cache' devices. *While an L2ARC read cache will speed up reads in most use cases, the SLOG only speeds up synchronous writes*.

The ZIL caches writes to guarantee their completion in the case of a power failure or system crash. *The ZIL normally exists as part of the ZFS pool, but with a SLOG, it resides on a separate, dedicated device. This speeds up performance by batching data together for synchronous writes for more efficiency*. These performance gains help with database operations, NFS operations such as virtualization where the operating system explicitly requests synchronous writes. If you aren't using something that is known to use synchronous writes like NFS or databases, chances are your SLOG will not help performance. A potential solution here is to set your pool to "sync=always". This ensures that every write goes to the write cache, improving write performance.

Too Many Snapshots

Snapshots give users the ability to rollback to previous system states to retrieve lost files or go back to a configuration that worked properly, while only saving the file system's blocks that have changed since the last snapshot. This results in near instant snapshot tasks. Snapshot tasks can be set for regular intervals and stay stored as long as desired.

While ZFS generally boasts that you can save unlimited snapshots, there are some practical limits to this. Some users may decide to have periodic updates every few minutes for multiple datasets and make their lifetime indefinite. Taking one snapshot every five minutes will require over 100,000 snapshots each year, creating some substantial performance loss. If you have thousands of snapshots, this means you will have thousands of blocks accumulating. Depending on the capacity of the disk, this can cause slowdowns when you list snapshots, possibly across the entire ZFS pool.

Upgrading your FreeNAS version with a full boot device

FreeNAS makes upgrading to the latest version, switching between nightly and release versions and rolling back to earlier versions very easy by storing snapshots of the OS on your boot device. *However, if you fill your boot device beyond its capacity, updating your OS version may result in the upgrade process mysteriously failing*. Fortunately, FreeNAS will give you an alert when your boot device exceeds 80% capacity, so you should know when your boot drive is getting full and deleting version snapshots is easy to do.

Just go into your System>>Boot tab and select the image you would like to delete and click on the delete button on the bottom of the page.

Rebuilding your ZFS array incorrectly

FreeNAS gives users the ability to set up ZFS arrays and resilver disks in the case of a drive failure. *If you remove the wrong disk and try to rebuild, you can end up losing your entire pool.* It is important to remember that the physical arrangement of the drives on your hardware may not correspond to your device numbers (ada0, ada1, ada2, etc.). To counter this, we recommend writing down the serial numbers for each disk along with which slot they're in, as the GUI will give you associated serial numbers in the case of a drive failure.

In addition, if you try to rebuild a ZFS array with a disk that is too small, your rebuild will fail. This can happen if you use a smaller capacity drive, say a 2TB instead of a 3TB, but it can also happen between different drives of the same listed capacity. Different drive manufacturers may create each drive with a slightly different total capacity, making the effective capacity of your replacement drive slightly higher or lower than the disk you replaced. If the capacity is slightly higher, your rebuild will succeed, but if it is slightly lower, it will not. *If a failure occurs on drives with the same listed capacities, there is a workaround available from the FreeNAS web user interface. Just access your system>>advanced menu and temporarily change your Swap Size to 0 before rebuilding*. Once your rebuild is complete, make sure to change it back, though (usually the default of 2GiB). The extra 2GiB should accommodate any small difference in drivef capacity but do try to use identical drives whenever possible.

Other Issues to Watch For

There are a couple of common issues with Active Directory that can cause problems. The first is if the system clock is out of sync. Make sure you're using a time server as AD/CIFS is very time sensitive. Second, having the domain name entered incorrectly can cause your Active Directory to have big problems. Ideally, your domain should have a reverse DNS entry, which you can determine easily enough: *https://www.google.com/search?q=dns+reverse+lookup&ie=utf-8&oe=utf-8#q=reverse+dns.*

Also, whenever possible, try not to mix sharing services on the same dataset. Differences in permissions between Unix (NFS) and Windows (CIFS) sharing formats can create some conflicts, so try to avoid this when you can. If you need users from multiple operating systems to have access to the same datasets, CIFS/SMB is your best choice. If you need to have multiple sharing protocols, you will want to separate your datasets between NFS & CIFS/SMB.

Finally, filling your storage pool over 80% of capacity will cause degraded performance. Try to plan your storage pool size to accommodate for this.

Conclusion

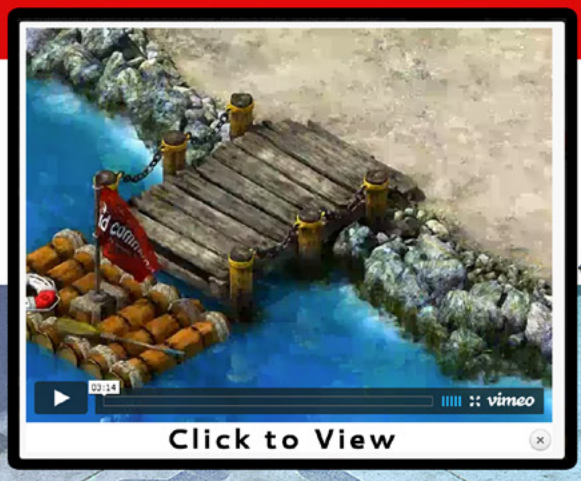
When deploying any server or storage system, setting up your system properly can help prevent headaches and even catastrophes down the road. As they say, an ounce of prevention is worth a pound of cure. While there are many aspects to setting up any given use case, this guide should avoid most of the major pitfalls people run into while setting up their FreeNAS storage. And if you're looking for even greater assurance, visit *www.ixsystems.com/truenas*, call us at 1-855-GREP-4-IX or email us at sales@ixsystems.com, for information on our qualified, professionally supported TrueNAS appliances. We look forward to hearing from you!

ABOUT THE AUTHOR

Mark VonFange has been working with iXsystems since 2008. He helps with first response support for Professional Services. He develops content for FreeBSD, PC-BSD, FreeNAS and Open Source and has been published in multiple technical publications.

ISO

mobile · interactive · design



- Mobile Apps
- Website Design
- Specialty Programming
- 3D Simulations
- Unity 3D
- SmartFoxServer
- Games
- Web & Database Dev
- Super friendly :)

reach out & let's talk: troy@isointeractive.com

www.isointeractive.com

The Cloud is as Secure as You Make It

BY DAN SREBNICK

Our cloud is secure. This is a statement that I hear over and over again from sales teams and pre-sales tech resources. Customers have heard it too, and they are not necessarily satisfied. Allow me to interpret.

All of the major operating systems providers tout the security of their operating systems. Microsoft, Apple, Oracle and the open source community all have strong arguments to make about the security built into their offerings.

However, no IT professional worth their salt would take such a simplistic view of security and stop there. Nor would they employ an enterprise or public facing system without enhancing the security capabilities of the host operating system environment with additional layers of protection providing visibility to the owner. This is what we refer to as defense-in-depth.

Ask your cloud provider HOW you are secure. Evaluate whether their explanation provides assurance of a level of security commensurate with the risk your organization is willing to take with regard to confidentiality, integrity and availability. The security focus of the cloud infrastructure provider is going to be to protect their shared infrastructure. Fill in the gaps with other cloud products or approaches to mitigate the risk to your application and data.

You would not likely build a mission critical application and place it on the public internet with a VLAN and some ACLs and expect it to last very long. Just like your datacenter, the cloud is as secure as you make it. Have a real security design for your cloud environment. Do your part.

And sales folks, be prepared to talk about what your security model addresses and what layers the customer might want to add to the platform. Whether it is alerting, log management, inline application threat mitigation, or a myriad of other security services that are available, be aware and partner with those service providers that add to your basic solution. Do your part too and it will add to your success.

ABOUT THE AUTHOR

Information security and information technology strategist seeking interesting projects and new challenges. I offer many years of experience in large scale program development, project management and operational oversight and will help to position your organization to defend itself against the cyber onslaught.

The Cloud It Self Is Not The Risk...

BY DAN SREBNICK

I spent a fair amount of the month of September on the road. I was talking to IT folks about the cloud. Specifically, one of my clients markets a cloud service to their customers. Their sales team had been hearing comments such as, "Our security folks would never agree to move to the cloud."

As NYC CISO, I realized early on that fighting the cloud was pointless. There are excellent use cases and business drivers for cloud use. So I embraced the cloud, in that I was open to exploring use cases that are right for the cloud.

It is a fallacy to assume that just because an asset does not sit in your data center that it is less secure than an asset in someone else's. I've seen security done poorly in my own data center and I've seen it done well in the cloud. It is all about risk management.

To take a risk management approach to the cloud, start with the classification of the data. (You do classify your data, don't you?) Determine the controls that would need to be implemented to adequately protect that data. Then go find a provider that either allows, or even better, will help you implement those controls in their cloud.

Find a cloud provider that will allow you to perform host and application vulnerability scans on your cloud assets. Many will. Coordination will likely be required, but then if you were doing the same testing in your own data center, you would hopefully have a notification methodology in case of impact.

And don't forget about an exit strategy. Have a migration plan in place that allows you to move your applications and data out of the cloud should requirements change.

Start small. Before migrating critical email or applications to the cloud, consider using a cloud provider as off-site backup storage. Or, find the most business critical application within your company that does not have an adequate disaster recovery plan and build an instance in the cloud.

Involve your security team in the discussion. Also involve other key executives. Talk about the business drivers, the risks, and the benefits. Take a rational approach and those clouds might look less ominous.

ABOUT THE AUTHOR

Information security and information technology strategist seeking interesting projects and new challenges. I offer many years of experience in large scale program development, project management and operational oversight and will help to position your organization to defend itself against the cyber onslaught.

Patterns For Cloud Integration:

Synchronous Vs Asynchronous Application Level Integration

MOHAMED FARAG, SUMMA TECHNOLOGIES

Recent statistics show that 90% of businesses have adopted at least one cloud application. 56% of enterprises are still identifying IT operations that are candidates for cloud hosting [1]. However, a recent survey, that was conducted by IDG Enterprise across 1600 IT decision makers, reflects that 46% of survey participants consider cloud integration as one of the major disconnects that hold organizations from going to the cloud [2].

What you will learn…

- The importance of cloud integration.
- Technical considerations in cloud integration.
- Key features of synchronous and asynchronous cloud integration patterns.

What you should know…

- A good understanding of object oriented principles.
- A basic understanding of cloud infrastructure and cloud technologies.
- A basic knowledge of cloud delivery models such as Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS).

Architecture styles evolved significantly in the past decade and opened new doors for cloud technologies, tools, and strategies. Cloud services enabled a new process thinking on data aggregation, data replication, shareable business functions, distributed computing, and business partner integration. It drove us to think about NoSQL databases, SaaS improvements, and data migration strategies. However, cloud computing also brought a lot of topics to the table. These topics included network latency, identity management, data security, interoperability, mobile access levels, application monitoring, application connectivity, and Service Level Agreements (SLAs). Enormous research and millions of dollars were invested in this area with the premise that the cloud will pay for such costs. In fact, recent statistics reveal that the general trend among IT decision-makers continues with efforts in cloud integration. The main driver, in this decision, is the Increasing Return on Investments (ROI), along with vast improvements in service quality [2].

As a result, major software players, such as IBM and Microsoft, have realized the importance of extending their applications to the cloud and they have been offering cloud integration as a major key feature in extending the lifetime of their software. In the same context, other software players (For instance Dell) have started the development of cloud-only applications; this due to the cost of

cloud integration. This article discusses two major cloud integration patterns that can help in reducing the cost of such expensive processes and promote the performance of such applications. This discussion focuses on two cloud integration patterns: synchronous operation offered from Remote Procedure Call (RPC), and Asynchronous Messaging (AM). Both patterns are designed to achieve application-level integrity under certain conditions.

The following section describes each pattern individually with respect to its general use, pros and cons. There are two types of cloud integration that are included in this investigation:

1. Ground-to-Cloud integration: Here the application was developed in a non-cloud environment and we are trying to adopt it to the cloud.
2. Cloud-to-Cloud integration: Here the application targets a cloud environment only.

Please note that Cloud-to-Ground integration goes beyond the scope of this article.

Remote Procedure Call (RPC)

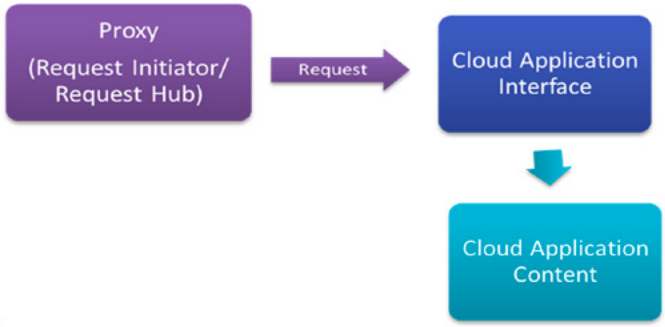


Figure 1. *RPC Cloud Integration Pattern Representation*

This pattern is used to integrate multiple applications so that they work together and can exchange information through each application's interface [3]. It is useful for information lookup in order share data among independent applications. In addition, this pattern is the ultimate solution when the data has to live with the source in a different area of the network. Furthermore, the use of an application interface promotes several key concepts such as encapsulation, abstraction, and interoperability.

Pros

1. Provides high reliability since it uses point-to-point communication by-default.
2. Ease of implementation as application integration pattern.
3. Data access at the source level.
4. Connects different independent applications, possibly running different technologies.

Cons

1. Synchronous operation. In other words, the caller is blocked until the operation is completed.
2. Lack of uniform security and transactional support.
3. Not suitable in large-scale cloud environments (large distributed environments).
4. Low Performance.
5. A high level of coupling between services since it assumes the availability of an existing service all the time.
6. Non-persisted data.
7. Limited commercial support.

There are on-going improvements to solve the challenges that are introduced by RPC. These improvements include the following topics:

1. Security: In this area, identity management can be used to enforce security in the communication.
2. Latency (Performance): There are several tips that can improve the performance over the network with respect to security such as:
   a. Acquiring authentication tokens (e.g. OAuth2).
   b. Callbacks and Caching.
   c. Increased the load of messages. In other words, avoid sending enormous number of small packets over the network.
3. Transactions: they are not supported by this pattern, so avoid using them for acceptable performance and right behavior.
4. Commercial Support: maintain communication to be HTTP oriented.

Now, how to use the value of this pattern in the extension of ground applications to cloud environment? There are general considerations when dealing with RPC patterns in ground-to-cloud integrations:

1. REST-Oriented.
2. Network Connectivity.
3. Identity Management.
4. Service Level Agreements.
5. Changing Schemas.

In order to account for these constraints and perform at the maximum levels, Table 1.0 shows possible implementation techniques that can mitigate significant risks.

Table 1. *Techniques for Ground-to-Cloud integration using RPC pattern*

Technique	Purpose	Implementation Example	Relative Complexity
Enterprise Service Bus (ESB), integration server	Used as middleware to manage the extension of ground application to the cloud	BizTalk Server, Mule ESB, RabbitMQ or Tibco ESB	Medium
Custom Code	Enforce point-to-point solutions	Java, .NET, Node.js	Varies

Admirably, cloud-to-cloud integration brings more consideration to the view. In fact, the considerations that would make sense in this context are:

1. Web Services.
2. Latency.
3. Service Level Agreements.
4. Monitoring.

For this set of considerations, the following techniques are available to overcome challenges associated with these considerations:

Table 2. *Techniques for Cloud-to-Cloud integration using RPC pattern*

Technique	Purpose	Implementation Example	Relative Complexity
Point-to-Point	Basic methodology for making integration not typically RPC	Custom Java or .NET Code	Varies
On-Premises Broker	You can extend it to relay although that is not efficient.	Custom code to build broker	Medium
Cloud hosted bus	Integration bus that is sitting in the cloud and managing communication between cloud endpoints.	Windows Azure Service Bus	High

The next subsection introduces asynchronous messaging integration pattern.

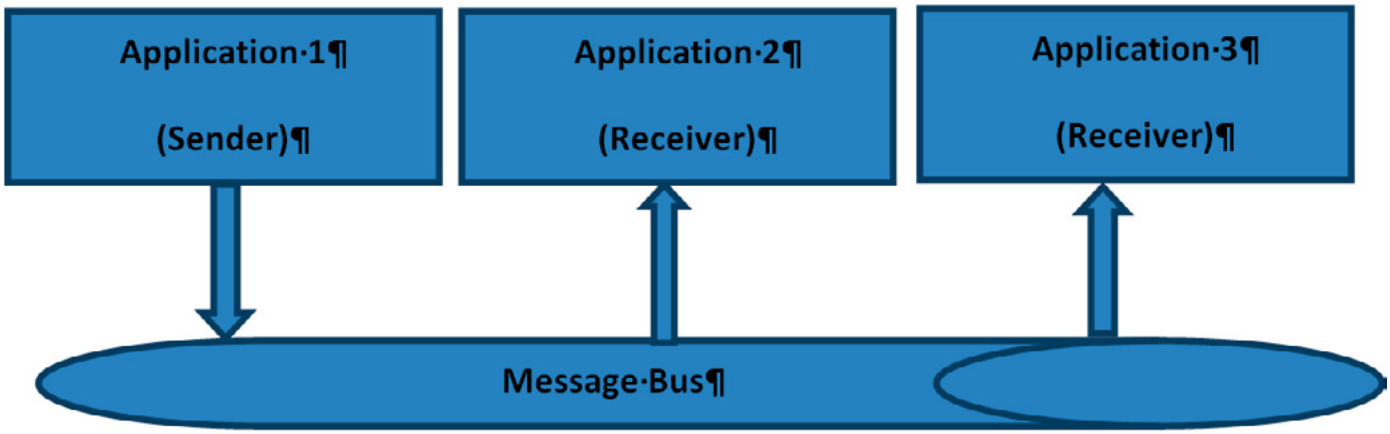


Figure 2. *AM Cloud Integration Pattern Representation*

Asynchronous Messaging (AM)

This pattern uses "Messaging" to transfer packets of data frequently, reliably, and asynchronously using customized formats [4]. This pattern is extremely useful for data sharing via broadcasted messages in which the caller does not have to be blocked during operation.

Pros

1. Callers are not blocked when making calls.
2. Ideal for broadcasting or multicasting.
3. Ideal for cloud-scale.
4. Can achieve higher reliability when brokers are used.
5. Embrace loose coupling.
6. Can be used for point-to-point or message routing to achieve content-based routing, message filtering, recipient list filtering, and aggregators.
7. Can function in stateful or stateless modes.

Cons

1. Not real-time synchronization. In other words, not consistent enough to manage the communication between modules that have some sort of dependency.
2. Achieving reliability may require store + forward which degrade the overall performance.
3. Idempotence often needed because of the possibility of message duplication.
4. Broadcasting requires parallelization due to the enormous number of messages that are received from peers.
5. Difficult to debug and trace.
6. Limited commercial application support.

Considering these advantages and limitations, how asynchronous messaging can be useful in Ground-to-Cloud integration?

Asynchronous Messaging is a great way to limit coupling and module dependencies. However, there are a few considerations to implement this pattern in Ground-to-Cloud integration:

1. Network Connectivity.
2. Message Monitoring.
3. Data Security.
4. Interoperability.
5. Destination System Capabilities.

The use of brokers is significant in the performance and reliability of this pattern. For example, brokers may boost the performance of the overall application with asynchronous push notifications that will promote caching the data that is frequently used. There are a few techniques that can be used to maximize the gain from Asynchronous Messaging, given the Ground-to-Cloud considerations such as those stated in Table 3.

On the other hand, how Asynchronous Messaging improve Cloud-to-Cloud integration?

Cloud-to-Cloud integration emphasizes several cloud topics including:

Table 3. *Techniques Ground-to-Cloud integration using AM pattern*

Technique	Purpose	Implementation Example	Relative Complexity
Asynchronous Web Service Operation	Implement basic asynchronous operations	Mule ESB, BizTalk Server or Custom Code	Medium
Queue	Good for managing durability	AWS Simple Queue Service	Medium
Message Broker	Managing complex scenarios	Windows Azure Service Bus Notification Hubs	High

Table 4. *Techniques for Cloud-to-Cloud integration using AM pattern*

Technique	Purpose	Implementation Example	Relative Complexity
Asynchronous Web Service Operation	Implement basic asynchronous operations	Windows Azure BizTalk services	Medium
Queue	Good for managing durability	Amazon SQS	Medium
Message Broker	Managing complex scenarios	Windows Azure Service Bus	High

Table 5. *Use Cases for RPC vs. AM*

Use Cases	RPC	AM
Maximize Performance		✓
Maximize Reliability	✓	Reliability can be raised by Brokers
Cloud Scalability		✓
Loose Coupling		✓
Transactions		Not preferred but can be used with cautious to idempotency
Low Bandwidth Network	✓	AM can be used if it switches to point-to-point communication mode
Content Based Routing		✓
Allowing User action during operation		✓
Ease of Implementation	✓	
Ease of Debugging and Tracing	✓	

1. Identity Management.
2. Different Service Level Agreements.
3. Message Monitoring.
4. Communication Management.
5. Interoperability.

There are some techniques mentioned in Table 4 that highlight these considerations.

In Summary

This article introduced two cloud integration patterns that are used to integrate applications. These patterns differ in their operational nature, although they achieve the same goal. In general, Asynchronous Messaging is more convenient for cloud purposes but there is no straight-forward answer to the "all-ages" pattern. Instead, an investigation into the situational use weighs heavily in the argument for one pattern versus the other. Table 5 shows sample use cases for each pattern:

Bibliography

- *http://www.forbes.com/sites/louiscolumbus/2014/11/22/cloud-computing-adoption-continues-accelerating-in-the-enterprise/*, spotlights from the study on cloud impact.
- *http://www.idgenterprise.com/report/idg-enterprise-cloud-computing-study-2014*, details on the study.
- *http://www.enterpriseintegrationpatterns.com/*, Enterprise Cloud Integration Patterns
- *https://www.mulesoft.com/resources/esb/cloud-integration-patterns*, Cloud Integration Patterns.
- *http://www.cloudcomputingpatterns.org/*, Cloud Computing Patterns

References

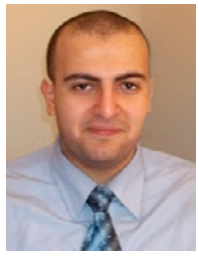
[1] *http://www.forbes.com/sites/louiscolumbus/2014/11/22/cloud-computing-adoption-continues-accelerating-in-the-enterprise/*, spotlights from the study on cloud impact.

[2] *http://www.idgenterprise.com/report/idg-enterprise-cloud-computing-study-2014*, details on the study.

[3] *http://www.enterpriseintegrationpatterns.com/patterns/messaging/EncapsulatedSynchronousIntegration.html*, details on RPC pattern.

[4] *http://www.enterpriseintegrationpatterns.com/patterns/messaging/Messaging.html*, details on Asynchronous Messaging pattern.

ABOUT THE AUTHOR



Mohamed Farag has been working in the IT field for six years. For two and half years, he has been working in software consulting and has acquired the following certifications:

- *Salesforce.com Certified Force.com Developer.*
- *IBM Certified Mobile Application Developer Worklight.*
- *Microsoft Certified Technology Specialist.*
- *CISCO CCNA Academy.*

On part-time basis, he pursues PhD in Systems Engineering at Colorado State University. Also, he has volunteered to review research papers for the International Journal of Computer Science and Information Technology (IJCSIT) since 2013. His detailed profile can be found at https://www.linkedin.com/in/mohamedsobhyfarag

Dr.WEB®
since 1992

The ERA of harmony and security

New Dr.Web! version 10

- Brand new user interface
- Configuration as simple as ABC
- Honest protection against real threats

Comprehensive protection for Windows
Anti-virus for Mac OS X and Linux

Basic protection for Windows,
Mac OS X and Linux

Protection for any and all devices*

Home and office

New purchase and renewal +150

* PC, Mac and mobile devices running OS supported by Dr.Web.

Protection for mobile devices — for free!

Doctor Web is the Russian developer of Dr.Web anti-virus software. Dr.Web anti-virus software has been developed since 1992. Doctor Web is one of the few anti-virus vendors in the world to have its own technologies to detect and cure malware. Dr.Web anti-virus software allows IT environments to effectively withstand any threats, even those not yet known.

Cloud Service in a Developer Point of View

DAVID CARLIER

This article will be an overview of writing a cloud service. Various ways exist to achieve your goals but we will focus on one that is memory efficient, multiplatform (POSIX systems), multi language (from C++ to Erlang), and reasonably fast. It is Apache Thrift. I recently fully wrote a cloud service and it worked reliably.

To illustrate this, we will make a basic remote file handler, the server is written in C++ and the client written in Python as an example.

Describing the service

Our server will be able to deliver three different services, listing files or directories, deleting or moving a file. Thrift is an IDL (Interface Definition Language) based framework, hence you describe your service via an abstract generic language and the Thrift compiler will generate the necessary code per programming language. The basic Thrift types are all we find in common in all languages, byte, binary, integer (i16/32/64), double, boolean, string, some containers as hashmap, sets or lists. For those familiar with C and or C++ we can define an atomic file with a “struct”:

```
struct file {
        1: string name
}
```

The number means the index of the name’s field. A file in a UNIX system can have several types, not necessary a regular file but a device, a socket and so forth. So, let’s enumerate each type we might need to identify the files, again “a la” C/C++:

```
enum file_type {
        FILE = 0,
        DEVICE = 1,
        SOCKET = 2,
        SYMLINK = 3,
        DIRECTORY = 4
}
...
struct file {
        1: file_type type
        2: string name
}
```

What if we store some file attributes like the size, the permissions bits ... ? Thrift allows to set a struct inside a struct without problems as you can see below:

```
struct file_attribute {
        1: i32 uid
        2: i32 gid
        3: i16 mask
        4: i64 size
        5: string strmask
}
struct file {
        1: file_type type
        2: file_attribute attr
        3: string name
}
```

Now, we can start to describe the three Thrift "services" as below, for the first we would like to return a map of files and for the sake of shortening, we "typedef" it as below:

```
typedef map<string, file> file_list
```

In addition, for our services, we would like to throw an exception in case something goes wrong. A Thrift exception is very similar to a struct:

```
exception file_exception {
        1: i16 code
        2: string msg
}
```

If we do not write the required keyword, a field is then optional. If you're not sure for future development that a field ought to be required, I'd suggest to leave it optional as the clients would stop working if the previous required field was suddenly optional in the server's side ...

```
service file_service {
        file_list eforensics_ls(1: required string path)
    throws (1: file_exception ex),
        i16 eforensics_rm(1: required string path) throws
    (1: file-exception ex),
        i16 eforensics_mv(1: required string src, 2:
    required string dst) throws (1: file_exception ex),
}
```

Above all of that, we might need to customize the language namespace to organize and avoid conflicts, for Java and C++ developers, for example, it is pretty well known. The namespace will be translated as well in the target language's logic:

```
namespace cpp eforensics.cloud
namespace py eforensics.cloud
```

The first will produce the usual C++'s namespace as

```
namespace eforensics { namespace cloud {
```

Listing 1.

```
namespace cpp eforensics.cloud
namespace py eforensics.cloud

enum file_type {
    FILE = 0,
    DEVICE = 1,
    SOCKET = 2,
    SYMLINK = 3,
    DIRECTORY = 4
}

struct file_attribute {
    1: i32 uid
    2: i32 gid
    3: i16 mask
    4: i64 size
    5: string strmask
}

struct file {
    1: file_type type
    2: file_attribute attr
    3: string name
}

typedef map<string, file> file_list

exception file_exception {
    1: i16 code
    2: string msg
}

service file_service {
    file_list eforensics_ls(1: required string path)
    throws (1: file_exception ex),
    i16 eforensics_rm(1: required string path) throws
    (1: file_exception ex),
    i16 eforensics_mv(1: required string src, 2:
    required string dst) throws (1: file_exception ex),
}
```

whereas the latter will make the eforensics/cloud Python module.

In the end, the Thrift file might look like this: Listing 1.

Generating the code

Once the service is defined, we can now use the Thrift compiler like this:

```
$ thrift -gen cpp eforensics.thrift
$ ls
eforensics.thrifts gen-cpp
$ thrift -gen py eforensics.thrift
...
```

In the C++ version, we realize that a skeleton server was generated as well, and we will use it to implement our service! (see Listing 2). Now it is up to us to implement the three services. Let's start with the simplest, removing a file with the famous C function unlink.

```
nt16_t eforensics_rm(const std::string& path) {
    if (unlink(path.c_str()) == -1) {
        return -1;
    }

    return 0;
}
```

Listing 2.

```
// This autogenerated skeleton file illustrates how to
   build a server.
// You should copy it to another filename to avoid over-
   writing it.

#include "file_service.h"
#include <thrift/protocol/TBinaryProtocol.h>
#include <thrift/server/TSimpleServer.h>
#include <thrift/transport/TServerSocket.h>
#include <thrift/transport/TBufferTransports.h>

using namespace::apache::thrift;
using namespace::apache::thrift::protocol;
using namespace::apache::thrift::transport;
using namespace::apache::thrift::server;

using boost::shared_ptr;

using namespace ::eforensics::cloud;

class file_serviceHandler: virtual public file_serviceIf {
 public:
  file_serviceHandler() {
    // Your initialization goes here
  }

  void eforensics_ls(file_list& _return, const
   std::string& path) {
    // Your implementation goes here
    printf("eforensics_ls\n");
  }

  int16_t eforensics_rm(const std::string& path) {
    // Your implementation goes here
    printf("eforensics_rm\n");
  }

  int16_t eforensics_mv(const std::string& src, const
   std::string& dst) {
    // Your implementation goes here
    printf("eforensics_mv\n");
  }

};

int main(int argc, char **argv) {
  int port = 9090;
  shared_ptr<file_serviceHandler> handler(new file_ser-
   viceHandler());
  shared_ptr<TProcessor> processor(new file_
   serviceProcessor(handler));
  shared_ptr<TServerTransport> serverTransport(new
   TServerSocket(port));
  shared_ptr<TTransportFactory> transportFactory(new
   TBufferedTransportFactory());
  shared_ptr<TProtocolFactory> protocolFactory(new TBi-
   naryProtocolFactory());

  TSimpleServer server(processor, serverTransport,
   transportFactory, protocolFactory);
  server.serve();
  return 0;
}
```

To improve it, we could make sure the file is a regular file, otherwise, return the exception we set earlier in the Thrift IDL file (see Listing 3).

In a similar manner, we can do the move function: Listing 4.

Then the last service, listing files or directories. Previously, we defined several types of files and their attributes, hence we'll once again rely on the stat function: Listing 5.

We are nearly done, let's compile the code.

```
$ c++ -std=c++11 -g -O2 -I. -I/usr/local/include -o efo-
    rensics_constants.o -c eforensics_constants.cpp
$ c++ -std=c++11 -g -O2 -I. -I/usr/local/include -o efo-
    rensics_types.o -c eforensics_types.cpp
$ c++ -std=c++11 -g -O2 -I. -I/usr/local/include -o file_
    service.o -c file_service.cpp
```

Listing 3.

```
void create_stat(struct stat &s, const string &path) {
      if (path.size() > MAXPATHLEN) {
          string msg = "Name too long ";
          msg += path;
          f.__set_code(-1);
          f.__set_msg(msg);
          throw f;
      }

     ::memset(&s, 0, sizeof(s));

      if (stat(path.c_str(), &s) == -1) {
          string msg = "Could not stat ";
          msg += path;
          f.__set_code(-1);
          f.__set_msg(msg);
          throw f;
      }
  }


....
struct stat s;
    create_stat(s, path);
    mode_t m = s.st_mode;

    if ((m & S_IFMT) != S_IFREG) {
        string msg = "Only files can be removed";
        f.__set_code(-1);
        f.__set_msg(msg);
        throw f;
    }

    if (unlink(path.c_str()) == -1) {
        string msg = "Could not remove ";
        msg += path;
        msg += ": ";
        msg += strerror(errno);
        f.__set_code(-1);
        f.__set_msg(msg);
        throw f;
    }
...
```

Listing 4.

```
int16_t eforensics_mv(const std::string& src, const
   std::string& dst) {
    struct stat s;
    create_stat(s, src);
    mode_t m = s.st_mode;

    if ((m & S_IFMT) != S_IFREG) {
        string msg = "Only files can be moved";
        f.__set_code(-1);
        f.__set_msg(msg);
        throw f;
    }

    if (rename(src.c_str(), dst.c_str()) == -1) {
        string msg = "Could not move ";
        msg += src;
        msg += " to ";
        msg += dst;
        msg += ": ";
        msg += strerror(errno);
        f.__set_code(-1);
        f.__set_msg(msg);
        throw f;
    }

    return 0;
  }
```

Listing 5.

```
int16_t ls_add_entry(file_list & _return, const string
   path) {
      bool isDir = false;
      struct stat s;
      create_stat(s, path);
      mode_t m = s.st_mode;

      file fl;
      fl.attr.uid = s.st_uid;
      fl.attr.gid = s.st_gid;
      fl.attr.size = s.st_size;
      fl.name = path;

      if ((m & S_IFMT) == S_IFBLK || (m & S_IFMT) ==
   S_IFCHR ||
                  (m & S_IFMT) == S_IFIFO) {
          fl.type = file_type::type::DEVICE;
      } else if ((m & S_IFMT) == S_IFSOCK) {
          fl.type = file_type::type::SOCKET;
      } else if ((m & S_IFMT) == S_IFLNK) {
          fl.type = file_type::type::SYMLINK;
      } else if ((m & S_IFMT) == S_IFREG) {
          fl.type = file_type::type::FILE;
      } else if ((m & S_IFMT) == S_IFDIR && (((m & S_
   IFMT) & S_IFLNK) != S_IFLNK)) {
          fl.type = file_type::type::DIRECTORY;
          isDir = true;
      }

      fl.attr.mask = 0;
      if (m & S_IWUSR)
          fl.attr.mask |= 0x400;
      if (m & S_IRUSR)
          fl.attr.mask |= 0x200;
      if (m & S_IXUSR)
          fl.attr.mask |= 0x100;
      if (m & S_IWGRP)
          fl.attr.mask |= 0x040;
      if (m & S_IRGRP)
          fl.attr.mask |= 0x020;
      if (m & S_IXGRP)
          fl.attr.mask |= 0x010;
      if (m & S_IWOTH)
          fl.attr.mask |= 0x004;
      if (m & S_IROTH)
          fl.attr.mask |= 0x002;
      if (m & S_IXOTH)
          fl.attr.mask |= 0x001;
      char strmask[5];
      sprintf(strmask, "%04x", fl.attr.mask);
      fl.attr.strmask = string(strmask);

      _return[path] = fl;

      if (isDir) {
          DIR *dir = opendir(path.c_str());
          if (dir == NULL) {
              return -1;
          }

          struct dirent entry, *result = NULL;
          // We could have just used readdir but we
      might need to run it
          // in multi thread context ...
          while (readdir_r(dir, &entry, &result) == 0) {
              if (result == NULL)
                  break;
              if (strcmp(".", result->d_name) == 0 ||
                  strcmp("..", result->d_name) == 0)
                  continue;
              string rpath = path;
              if (rpath[path.size() -1] != ,/')
                  rpath += "/";
              rpath += result->d_name;
              ls_add_entry(_return, rpath);
          }

          closedir(dir);
      }

      return 0;
    }
...

// It is better in terms of interface, in the case of
   C++, to do not return
// a map as the IDL defined
void eforensics_ls(file_list& _return, const std::string&
   path) {
      _return.clear();

      ls_add_entry(_return, path);
}
...
```

Listing 6.

```
$ ./file_service-remote -h 192.168.1.11:9090 eforensics_ls /tmp
{ ,/tmp': file(type=4, attr=file_attribute(gid=0, mask=None, uid=0, strmask='0777', size=None), name='/tmp'),
  ,/tmp/.ICE-unix': file(type=4, attr=file_attribute(gid=0, mask=None, uid=0, strmask='0777', size=None), name='/tmp/.
   ICE-unix'),
  ,/tmp/.ICE-unix/1997': file(type=2, attr=file_attribute(gid=1000, mask=None, uid=1000, strmask='0777', size=None),
   name='/tmp/.ICE-unix/1997'),
  ,/tmp/.X0-lock': file(type=0, attr=file_attribute(gid=0, mask=None, uid=0, strmask='0222', size=None), name='/tmp/.
   X0-lock'),
  ,/tmp/.X11-unix': file(type=4, attr=file_attribute(gid=0, mask=None, uid=0, strmask='0777', size=None), name='/tmp/.
   X11-unix'),
  ,/tmp/.X11-unix/X0': file(type=2, attr=file_attribute(gid=0, mask=None, uid=0, strmask='0777', size=None), name='/
   tmp/.X11-unix/X0'),
  ,/tmp/.vbox-dcarlier-ipc': file(type=4, attr=file_attribute(gid=1000, mask=None, uid=1000, strmask='0700',
   size=None), name='/tmp/.vbox-dcarlier-ipc'),
  ,/tmp/.vbox-dcarlier-ipc/ipcd': file(type=2, attr=file_attribute(gid=1000, mask=None, uid=1000, strmask='0700',
   size=None), name='/tmp/.vbox-dcarlier-ipc/ipcd'),
  ,/tmp/.vbox-dcarlier-ipc/lock': file(type=0, attr=file_attribute(gid=1000, mask=None, uid=1000, strmask='0600',
   size=None), name='/tmp/.vbox-dcarlier-ipc/lock'),
  ,/tmp/config-err-tu3hNl': file(type=0, attr=file_attribute(gid=1000, mask=None, uid=1000, strmask='0600', size=None),
   name='/tmp/config-err-tu3hNl'),
  ,/tmp/unity_support_test.0': file(type=0, attr=file_attribute(gid=1000, mask=None, uid=1000, strmask='0662',
   size=None), name='/tmp/unity_support_test.0')}
```

Listing 7.

```
...
if http:
  transport = THttpClient.THttpClient(host, port, uri)
else:
  socket = TSSLSocket.TSSLSocket(host, port,
   validate=False) if ssl else TSocket.TSocket(host,
   port)
  if framed:
    # In this mode, the message is fully read no flush is
   required
    transport = TTransport.TFramedTransport(socket)
  else:
    transport = TTransport.TBufferedTransport(socket)
protocol = TBinaryProtocol.TBinaryProtocol(transport)
client = file_service.Client(protocol)
transport.open()
...
# Pretty straightforward to call each server method as
   you can see
if cmd == ,eforensics_ls':
  if len(args) != 1:
    print(,eforensics_ls requires 1 args')
    sys.exit(1)
  pp.pprint(client.eforensics_ls(args[0],))

elif cmd == ,eforensics_rm':
  if len(args) != 1:
    print(,eforensics_rm requires 1 args')
    sys.exit(1)
  pp.pprint(client.eforensics_rm(args[0],))

elif cmd == ,eforensics_mv':
  if len(args) != 2:
    print(,eforensics_mv requires 2 args')
    sys.exit(1)
  pp.pprint(client.eforensics_mv(args[0],args[1],))

else:
  print(,Unrecognized method %s' % cmd)
  sys.exit(1)

transport.close()
```

```
$ c++ -Wall -std=c++11 -g -O2 -I. -I/usr/local/include -o
   file_service_server.o -c file_service_server.skeleton.cpp
$ c++ -std=c++11 -g -O2 -o eforensics_file_service efo-
   rensics_constants.o eforensics_types.o file_service.o
   file_service_server.o -Wl,-rpath,/usr/local/lib -L/usr/
   local/lib -lthrift
```

If you execute the final executable, it will listen via the 9090 port and if you generated Python's version, for example, it should have generated a sample client: Listing 6.

We can have a quick look at how the Python's version is made: Listing 7.

If we come back to the C++ server's code, the skeleton's generated code uses a TsimpleServer which is perfect for start but is monothread. I'd suggest the TThreadPoolServer (more efficient than the TThreadedServer) or the TNonBlockingServer instead and to at least add a signal handler to terminate the server properly. The TthreadPoolServer's version might look like this: Listing 8.

Conclusion

Apache Thrift works well indeed in most POSIX systems, I've made the full example server part in a Linux machine and tested with FreeBSD and Linux. The client was called on a remote FreeBSD's machine.

There exists an alternative version remade by Facebook called fbthrift which works fully only on Linux but the code generated is superior and this version in general has proved to be more efficient in terms of memory usage at least. There also exists Google Protocol Buffer which performs better than the two above and has less languages supported (officially). You have to write the client / server code on your own, though. So based on your own criteria and restrictions, one of these might fit better for your own case.

ABOUT THE AUTHOR

David Carlier is a developer since 2001, mainly C/C++, living and working in Ireland mainly since 2012. He contributes to some open source projects and uses in a daily basis various operating systems mainly BSD flavours.

Listing 8.

```
...
  signal(SIGINT, servsighandler);
    signal(SIGQUIT, servsighandler);
    signal(SIGPIPE, servsighandler);

    try {
        shared_ptr<TProcessor> processor(new cloud_service_adminProcessor(handler));
        shared_ptr<TServerTransport> serverTransport(new TServerSocket(port));
        shared_ptr<TTransportFactory> transportFactory(new TBufferedTransportFactory);
        shared_ptr<TProtocolFactory> protocolFactory(new TBinaryProtocolFactory());

        threadManager = ThreadManager::newSimpleThreadManager(workers);
        shared_ptr<PosixThreadFactory> threadFactory(new PosixThreadFactory());
        threadManager->threadFactory(threadFactory);
        threadManager->start();

        std::clog << "server is starting" << std::endl;
        nserver = shared_ptr<TServer>(new TThreadPoolServer(processor, serverTransport, transportFactory, protocol-
   Factory, threadManager));
        nserver->serve();
    } catch (std::exception &e) {
        std::clog << "An error occured: " << e.what() << std::endl;
    }
```

Dr.WEB®
CureIt!™

EMERGENCY CURING

for Windows workstations and servers

including those running other anti-virus software

FUNCTIONS:

- Cures Windows workstations and servers.
- Verifies the quality of the anti-virus software currently in use.

FEATURES:

- Dr.Web CureIt! doesn't require installation and doesn't conflict with any known anti-virus; consequently there is no need to disable the anti-virus currently in use to check a system with Dr.Web CureIt!.
- Improved self-protection and an enhanced mode for more efficient countermeasures against Windows blockers.
- Dr.Web CureIt! is updated at least once an hour.
- The utility can be launched from removable media including USB storage devices.

LICENSING FEATURES:

The utility is available for free when used for non-business purposes.

Doctor Web is the Russian developer of Dr.Web anti-virus software. Dr.Web anti-virus software has been developed since 1992. Doctor Web is one of the few anti-virus vendors in the world to have its own technologies to detect and cure malware. Dr.Web anti-virus software allows IT environments to effectively withstand any threats, even those not yet known.

Getting Started with Go on FreeBSD

BRIAN DOWNS

Two of my favorite things are the FreeBSD operating system and the Go programming language. The two are similar inasmuch as they're uniquely equipped to solve difficult problems in different ways from others in their respective categories. FreeBSD and Go together yield a powerful combination for productivity and fun.

To get started, we need to get Go installed. We'll be using a fresh installation of FreeBSD 10.2. There are a number of ways to install Go. We'll take a look at what's involved with two of the more obvious and popular methods. For the ports installation, Go can only be compiled and installed on i386, amd64, and armv6 systems.

Ports Installation

The first thought by anyone using a BSD distribution might be to go to the ports tree and find Go. This was my first thought as well, however, let's review the Makefile first. We find that this port (*http://www.freshports.org/lang/go*) is maintained by *jlaffaye@FreeBSD.org* and does some architecture checks prior to compilation and installation.

FreeBSD 10.2's ports tree is already updated to the latest (as of this writing) version of Go, 1.5.1. To successfully compile Go 1.5 (*https://blog.golang.org/go1.5*), we need Go 1.4 (*https://blog.golang.org/go1.4*) which is outlined on the dependency line of the Makefile.

To start the process, issue the command below.

```
$ make install clean
```

Since `go1.5.1.src.tar.gz` doesn't exist in `/usr/ports/distfiles/`, it will have to be downloaded. Once the download is complete, the compilation process begins. Go 1.4.3 is installed and used to build Go 1.5.1. All 1.5.1 files are placed in `/usr/local/go` and the 1.4.3 files are placed in `/usr/local/go14`.

Manual Installation

Manual installation is a bit less sexy and carries a little more weight since you have to manage the download, untarring, copying the files into place, and you're also not compiling the system. It's a pre-built binary compiled to run for your operating system and architecture.

```
$ wget https://storage.googleapis.com/golang/
   go1.5.1.freebsd-amd64.tar.gz
$ tar -C /usr/local -xzf go1.5.1.freebsd-amd64.tar.gz
```

Post Installation

No matter which installation process you've chosen, the following commands will have to be run to set-up your Go environment. More can be found here: *https://golang.org/doc/code.html*

```
$ mkdir -p ~/gocode/src ~/gocode/bin ~/gocode/pkg
$ mkdir -p ~/gocode/src/github.com/briandowns
$ export GOROOT=/usr/local/go
$ export PATH=$PATH:$GOROOT/bin
$ export GOPATH=/home/bdowns/gocode
```

Go expects the “src”, “bin”, and “pkg” directories to be created in the path. These directories hold source files, package objects, and compiled binaries respectively and are known as a Go Workspace (*https://golang.org/doc/code.html#Workspaces*). The “go” command expects to find these directories by using the environment variable GOPATH which is the top level of where those directories live.

To make sure that we have everything installed and environment variables set correctly, we should run a quick Go command. Checking the version will suffice. Run the command below:

```
$ go version
```

The expected output should be, “go version go1.5.1 freebsd/amd64”. If all of that checks out, we’re ready to write some code.

Our First Program

For our first program, we’re going to do the traditional “hello, world”. You’ll want to cd to your source control directory (i.e. `${GOPATH}/src/github.com/briandowns`) where you’ll need to create a new directory called “hello”. In the newly created “hello” directory, use your favorite editor and create a file named “main.go”. Add the text below to the file.

```
package main

import (
    “fmt”
)

func main() {
    fmt.Println(“hello, BSD!”)
}
```

Listing 1.

```
package main

import (
    “log”
    “os”

    “github.com/go-fsnotify/fsnotify”
)

var watchDir = “/usr/local/directory_of_interest”
var logPath = “/usr/local/var/log/dir_logger.log”

func main() {
    os.Exit(realMain())
}

func realMain() int {
    // open up our log file
    f, err:= os.OpenFile(logPath, os.O_RDWR|os.O_
    CREATE|os.O_APPEND, 0666)
    if err != nil {
        log.Println(err)
        return 1
    }
    // close the file cleanly when function exits
    defer f.Close()

    // set the log file as the logging endpoint
    log.SetOutput(f)

    // setup the new watcher
    watcher, err:= fsnotify.NewWatcher()
    if err != nil {
        log.Println(err)
        return 1
    }
    // make sure the watcher is closed cleanly
    defer watcher.Close()

    // start the goroutine and have it listen for events
    go func() {
        for {
            select {
            case event:= <-watcher.Events:
                log.Println(“event:”, event)
            case err:= <-watcher.Errors:
                log.Println(“error:”, err)
            }
        }
    }()

    // add the configured dir to the watcher
    err = watcher.Add(watchDir)
    if err != nil {
        log.Print(err)
        return 1
    }

    return 0
}
```

At this point, save and close the file. We can now either build the file or execute it without building. We have three choices of how to proceed.

```
# build and run from a temporary location
$ go run main.go
```

or

```
# build a binary and run
$ go build && ./hello
```

or

```
# build, move binary to bin dir and run
$ go install && rehash && hello
```

Congratulations! You've just written your first Go program on BSD. Not fulfilled? I know…. There are a lot of utilities in the ecosystem that could use an upgrade. A lot of them are written in C or C++ which is a lot more verbose and more difficult to maintain over time than Go code. So go rewrite something and see how it turns out.

FreeBSD Specific Applications

There are a number of great BSD specific applications, tools, and utilities being written in Go, however, I've found one to be extremely interesting and promising, too. JetPack (*https://github.com/3ofcoins/jetpack*) From the site, "Jetpack is an experimental and incomplete implementation of the *App Container Specification* for FreeBSD. It uses jails as isolation mechanism, and ZFS for layered storage."

An amazing package is fsnotify (*https://github.com/go-fsnotify/fsnotify*). It makes interacting with file system kernel events (kqueue) extremely simple. Below is an example that outputs an event when one happens in the configured directory (see Listing 1).

This is great if we want to execute it each time we're expecting a series of events we might want to watch for but it's not very sustainable. To ease this burden, a simple startup script can be added to `/usr/local/etc/rc.d` which uses the FreeBSD RC system (*https://www.free-*

Listing 2.

```
#!/bin/sh
#
# PROVIDE: dir_logger
# REQUIRE: syslog
# KEYWORD:

. /etc/rc.subr

name="dir_logger"
rcvar="dir_logger_enable"
command="/usr/local/bin/dir_logger"

dir_logger_user="root"

start_cmd="/usr/sbin/daemon -f -u $dir_logger_user $com-
    mand"

load_rc_config $name
: ${dir_logger_enable:=no}

run_rc_command "$1"
```

Listing 3.

```
package main

import (
    "flag"
    "fmt"
)

var (
    firstFlag string
    lastFlag string
    ageFlag int
)

func init() {
    flag.StringVar(&firstFlag, "f", "", "your first name")
    flag.StringVar(&lastFlag, "l", "", "your last name")
    flag.IntVar(&ageFlag, "a", 0, "your age")
}

func main() {
    flags.Parse()

    fmt.Printf("Hello, %s %s. Your are %d years old.\n",
   firstFlag, lastFlag, ageFlag)
}
```

bsd.org/cgi/man.cgi?query=rc.subr). To finalize this, just `dir_logger_enable="YES"` to the `/etc/rc.conf` file (see Listing 2).

A big part of building CLI applications is being able to parse command line arguments. Go comes with a really simple to use package called “flags” to help you do that. Below is an example of an application that will read the arguments given and print them back out (see Listing 3).

There are a number of packages to do a lot more with your command line parameters. The two most popular are *http://github.com/mitchellh/cli* and *http://github.com/codegangsta/cli*. Both of these packages offer great features worth looking at.

FreeBSD makes for a powerful development workstation. With an ample amount of packages providing the ability to get very low level with the system, an extremely simple installation and upgrade process, and simple syntax, Go is a great language to interact with and build applications for FreeBSD.

ABOUT THE AUTHOR

Brian Downs
briandowns.github.io
github.com/briandowns
@bdowns328

Interview with Brian Callahan

BY MARTA ZIEMIANOWICZ AND MARTA STRZELEC

Brian is a Ph.D. student in the Science and Technology Studies department at the Rensselaer Polytechnic Institute in Troy, NY.

BSD Magazine: Where did the idea of Devio.us came from? What it is about?

Brian Callahan: Devio.us began back in 2010 when some friends got together with the idea of starting a shell service. Being big fans of OpenBSD, they decided to use it as the base of their service, and Devio.us was born.

Devio.us offers a number of services for our users including personal web space, email, an IRC cloak, and even their own personal gopherhole! It is a special mix of retro and modern, blended into a culture and a community that is passionate about *BSD. Since opening in 2010, nearly 6,000 people have gotten accounts with just under 4,000 users still active.

Today, Devio.us has a unique mission: it is both a free OpenBSD-based shell service provider and an online *BSD user group. This mission is accomplished by our critical devotion to building and maintaining our community. We try to be a bridge between those coming to *BSD for the very first time and seasoned developers, and everyone in between. As a community, we are really proud of all the work our users have accomplished. A number of OpenBSD developers got their start on Devio.us, and we work tirelessly to have a community where people can come together regardless of where they are in their *BSD journey and foster an environment of encouraging *BSD development--from the four main *BSD projects to small side projects for inclusion into to ports trees.

Going forward, Devio.us is looking at a multi-dimensional expansion beyond just OpenBSD. The rebranding of the service, the addition of an online *BSD user group to our activities, coincided with our recent talk about the technical and social lessons learned running Devio.us at vBSDcon 2015. This means that we are actively increasing our advocacy to encompass all the BSDs. With the increased technical advocacy joins increased social diversity: Devio.us is hoping to become a model for inclusion in tech communities, not just in raw demographics but in understanding of why diversity matters on a technical and social level. We would also love to help make this a reality.

BSD Mag: That is really interesting – what kind of social lessons did you learn running Devio.us?

BC: The main lesson, which I use to end the vBSDcon talk, is to care less about your technology and more about your people. The best technology in the world will not create a community. But if you focus on creating a community in which everyone feels like they have ownership over it in some way, that can forgive even some bad technology. I think one of the primary reasons we have had to remove so few people is because our community understands that harming the server, the technology, does not just harm the admins but also themselves, their friends, and the whole community. Our users are our best policy enforcers. They understand that Devio.us is our collective home and they are willing to spend the time and energy protecting it.

BSD Mag: Is there is any philosophy behind Devio.us?

BC: Probably not so much at the beginning, but I only became an administrator in 2013! As I understand it, the philosophy in the early years was simply to show how great OpenBSD was and how easy it was to run a shell service using it.

Since joining the admin team, I have used Devio.us to think about what inclusion and diversity mean in open source and tech more broadly. This is certainly a reflection of my day job as a social scientist! So if we have any philosophy today, I would say it is to be a *BSD success story making other *BSD success stories and to be a space that is always reflecting on who we are missing out on, why they matter, and figuring out how to improve our community and ourselves in that regard.

BSD Mag: What do inclusion and diversity In tech and security mean to you? Do you think that those fields are more open than others, or is it about different criteria for inclusion?

BC: Open source still suffers from a gross lack of women and other minority voices. More so than the tech industry at large. The many initiatives to remedy this are awesome and often awe-inspiring. The beauty of seeking diversity is that you bring in people with vastly different experiences and skill sets, who can both see and fix problems that you cannot, as well as offer new perspectives to strengthen the code and the community.

In that regard, changing the demographics in and of themselves, while vitally necessary, is not the final step. One could easily imagine a scenario where diversity is done right "on paper" but nothing has changed where it fundamentally matters. So we have to get it right on paper as well as getting those diverse voices into situations and conversations that matter, so that all that expertise and experience is a part of the process and the product.

It is not always a popular opinion to have. But open source likes to talk about how it is open for everyone to participate. It is long past time to make that talk a reality. Devio.us should be a place where talk and action come together.

BSD Mag: Who are your users? What topics do they like the most?

BC: I would say we have a fairly typical user base, the one uniqueness is the dedication to the *BSD family of operating systems. Hopefully, that will change in the future.

BSD Mag: You've already mentioned expanding Devio.us beyond *BSD – can you tell us what direction will this take?

BC: We want to expand beyond OpenBSD, to include all the BSDs. All the *BSD user groups that have been around a decade or more have done so by being *BSD agnostic. We want Devio.us to have a nice long life, so becoming *BSD agnostic ourselves is one of the ways of doing that.

The best examples to point out are whenever a new OpenBSD snapshot is released, one of our bots in the #devious IRC channel will announce it. Also, every OpenBSD developer in the channel gets a shout out from the bot when they make a commit. Every *BSD developer who is a part of Devio.us--regardless of what project she or he works on--should get a shout out when a commit is made. And we should announce all the releases and snapshots for all the BSDs. That is one small way we can start being more *BSD agnostic. I am sure there will be more changes in the future, and we are appreciative to anyone who has ideas for how we can do things better.

BSD Mag: Your name- devious, where is it from? Who would you like to outsmart?

BC: Unfortunately, the story behind the domain name is not so interesting: the founders of Devio.us noticed that the domain was available and they thought it would be a fun domain name to have!

BSD Mag: You have many rules regarding community. Are users problematic?

BC: It looks like we have more rules than we really do. Everything can be reduced to two main rules: 1. do not leave a mess for the admins to clean up, and 2. try to be a part of the community. I think rule 2 is the more important rule. It is what has kept Devio.us around for as long as it has been and is why we have seen the community grow larger and stronger. Devio.us is not just some box that you can SSH into, run your IRC bouncer, and never think about again. We want users who will be invested in and take ownership of Devio.us. Our users care about the service and want to see the service grow with them. This is why we forbid things like IRC bouncers. We want people. Bots are not people.

In all of Devio.us, we have removed 177 users. And that number does not paint an accurate picture. Most of those 177 users emailed us asking if we would remove their account because they did not think they would be using it any more. We are always sad to see people go, but understand that some people want to leave. So when we get those requests, we do delete the account but it adds to the counter we keep of users removed.

I would guess that the actual number of users removed for breaking rules is quite low. Probably not more than 20.

BSD Mag: So they have to become friends with you first?

BC: That is one way to look at it. I think, though, the community as a whole sees it as welcoming someone new into the community before that person gets an account.

The interesting thing is that most newcomers on IRC who say they want an account get answered quickly not by an admin, but usually by a regular community member who will give the newcomer the broad overview of who we are and what we are about. The one thing this requirement, which is new, has done for us is cut down on the number of applications by people who do not want to be part of the community. They come to IRC, notice that we are a different kind of shell service, and leave without submitting an application because they realize that running a bot and leaving is not something they can do with us.

BSD Mag: Devio.us is for free. What do you think about open source?

BC: It is important that Devio.us always be available to our users for free. Access to a community can never be dependent on one's ability to pay: doing otherwise would run counter to our goals of technical and social inclusion.

As for open source, we love it! We would never be able to do Devio.us without it. Open source lets us focus on building our community and not have to worry about software suddenly breaking. Plus, with the six month release cycle of OpenBSD, and the binary updates available from M:tier's free service, we can be sure that not only will software not suddenly break but it is also receiving regular security updates. That protects us and our users. Devio.us is an open source, *BSD success story with a very exciting future. We hope you will join us!

BSD Mag: There is a note that you are not interested in any info about the users, but to make an account, you have to fill everything in, together with name and e-mail address, etc. So how does it work?

BC: We do ask for a few things on the sign-up form: your name, an email address for us to send you an auto-generated password should you get accepted for an account, your desired username, what default shell you want, how you heard about us, who you chatted with in IRC, and what you plan on doing with the account. This is mostly again to rule out those who just want a place to put an IRC bouncer.

I do understand that not everyone will feel comfortable entering their real name in the form. In that case, please reach out to me either by email or Twitter. We can always be accommodating for those who need. And if there are more ways we can make the process better, we want to hear that too!

BSD Mag: Give our readers your contact details!

BC: If you want to reach out to the admins, admins@devio.us.

Me, personally, I can be found at bcallah@devio.us or on Twitter @__briancallahan.

BSD Mag: Any thoughts or advice you would like to share with our audience?

BC: Short but sweet: don't forget to have fun, and don't be afraid to challenge yourself and be challenged by others. It's about the journey, not the destination.

BSD Mag: Thank you for talking with us.

BC: Thank you!

Thanks for interviewing me!

Brian Callahan

Brian is a Ph.D. student in the Department of Science and Technology Studies at Rensselaer Polytechnic Institute in Troy, NY. His research interests focus on the intersections of Open Source and Social Justice, include using Open Source software to teach STEM to underprivileged K-12 students and understanding and aiding the efforts to increase diversity and inclusion in Open Source. A former OpenBSD developer, Brian is involved in many facets of the *BSD community, including being a member of the admin group for the New York City *BSD User Group (NYC*BUG), the Capital District *BSD User Group (CDBUG), and the Devio.us shell provider, giving talks at various *BSD conferences, and teaching *BSD to undergraduate students at RPI.

Meet the **Developer-Friendly** Payment Solution

3 easy steps to optimized checkouts:

1

Create the checkout page

With Gate2Shop, you can optimize your payment pages by using ready-made templates or by customizing payment pages to your site look and feel.

2

Test and optimize

An effective payment page variant testing tool, A/B Testing helps you gain insight into user behaviour, increase payment conversion in the short and long term.

3

Accept payments worldwide

With dozens of alternative and local payment methods offered in multiple currencies, the personalized checkout allows you to reach users from all around the world.

✔ Easy integration ✔ Cross-platform ✔ Secure

G2S gate2shop
Sell. More.

Call for a free consultation: +44 20 3051 0330

www.g2s.com

Among certain sections of the marketing, editorial and certainly advertising communities, the use of Ad blockers is considered immoral, and in some cases users have been accused indirectly of theft. Are these users leeches or just more savvy netizins?

ROB SOMERVILLE

My dear father, who shuffled off this mortal coil some time ago, was a pragmatist. “Son”, he would say, “If there is something on television you don’t like there is always the off button”. Apart from being staunchly independent, his commitment to freedom of speech was unquestionable, but more to the point, his belief in the freedom of silence was as well.

One of the most irritating facets of modern life is the way that our personal boundaries are constantly under attack. Be it by marketeers, scammers or even hackers, the days of posting a “No circulars” sign next to your mailbox and expecting privacy are over. Irrespective of communications medium, be it postal, telephone, television, email, website or even the humble till or parking receipt, we are bombarded with the shrill but persistent siren call of those that want to grab our attention, or more accurately – our wallet. This deluge of disconnected imagery and words is rarely targeted creatively or effectively, despite the valiant attempts of advertisers to categorise by socio-economic class and the multitude of tools and data they have at their disposal. Hence the major growth in “loyalty cards”, a disingenuous term at best, as often the information gleaned from such data gathering exercises is sold off to other third parties. Woe betide the day when technology advances far enough for neural implants to monitor our emotions, something not too far fetched considering the implants already being developed [1].

The whole advertising issue boils down to one simple fact – privacy. As a sentient, unique individual, while I fully appreciate the need for developing brand identity and getting the message across about the efficacy or affordability of a product or service, at the same time I deeply value boundaries and my own personal space. An Englishman’s home is his castle, and in particular what seeps in from the outside world to my eyes and ears is important to me. If I am in the market for an electric spaghetti fork I will type the magic words into Google or Amazon. I don’t want to see that advert everywhere I go on the web. But it is brand *awareness* the marketeers will say is important, by continually bombarding our subconscious with XYZ the hope is that the particular product or manufacturer will float to the top when we are ever in need of their services.

To quote Edward Bernays, the father of public relations, the whole matter revolves around the engineering of consent. This somewhat sinister definition reeks on many counts, at the very least implying that consent was not present in the first place. It is no wonder that the term Public Relations was adopted to refer to this form of well researched social psychology rather than the more visceral and accurate title “Propagandist in Chief”.

If the truth be know then, mass-market advertising does work. Unfortunately though, it is based on the foundation of sleight of hand and manipulation. As adults, we are often only consciously aware of this fact intermittently, as the power of repetition has the uncanny knack of short-circuiting the conscious mind and travelling direct to our sub-

conscious. As we are then heading towards the area of the spiritual, this is very sensitive ground indeed. The pictures and associations I want in my inner psyche should be up to me, I should be allowed to choose to absorb or not as the case may be. However, when the blanket level of saturation reaches the point that we cannot ignore this intrusion, as individuals we hit critical mass and actively reject the message along with both baby and bathwater. Think compassion fatigue. And a totally counter productive experience for the advertiser as well.

Unfortunately for the marketing industry, consumers are becoming much more aware of the subtle and not so subtle techniques being used. A considerable percentage of the population are abandoning the traditional media (newspapers and television) for the vast expanse of the Internet, where the electric spaghetti fork enthusiast will find a specialist website dedicated to his or her needs. But alas, our enthusiast has an ad-blocker installed. How will they discover the latest model or find out about that neoprene-lined velvet accessory cover for their fork? By word of mouth and recommendation – undeniably the best form of advertising that there is. Here lies the conundrum behind advertising – it has no credibility other than sheer weight of presence. If I were an alien from the planet Zorg, I would be unable to make any decision regarding the credibility and integrity of a product against a competing item. However if I were to ask an earthling on a specialist forum what is the best fork, while it might result in a plethora of opinions, hopefully there would be some consensus amongst the group based on experiential data rather than just sheer hype. That is not to say that adverts are dishonest as such, rather the majority – by their sheer nature – are designed to be superficial and all encompassing so that they reach the widest possible audience. It is not until you get into the arena of high quality and very expensive advertising campaigns that you reach the point of actually admiring or appreciating the advertising. It is generally the shotgun, rather than the sniper rifle approach. And I resent being collateral damage.

Hence the wrath of those denied access to my eyes via my browser. While there could be argued that there is no technical solution to television advertising (personally I use the fast forward button and rarely watch live TV, only recordings), our PC's, laptops and mobile devices are considered much more personal and intimate devices. Of course there is the argument that by running an ad blocker I am denying the owner of the website of advertising revenue, but this is a specious argument. Even if I was not running an ad blocker I would not be clicking on the ad. So the only true beneficiary is not really the advertiser (who if they were that desperate to make their product or company appeal would be more engaged with the website owner, e.g. free evaluations or sponsorship) but the industry *behind* the advertising and the corresponding click-through. After all, most of the ad blockers work on denying access to the content delivery networks, and from a security perspective that is no bad thing considering the amount of malware and trackers that are attacking browsers these days. If an advertiser is so offended by ad blockers and alleged loss of revenue, why not put up a paywall and restrict access to paying visitors only. They would then quickly discover the real rather than the perceived value of their content. Rather than patronising and offending their audience, the marketeers would be better served finding creative ways of engaging with their target audience in a way that stimulates interest and debate, rather than trying to jam their foot in the gap and getting the technological door slammed in their face as a consequence.

So the gloves are off as far as the marketing sector is concerned and by choosing not to download their unsolicited click-bait, more and more sites are preventing access if they detect ad blockers in use. That's fine by me, for it demonstrates that they are more interested in selling to me than informing me, more concerned with catching my eye than engaging my mind, but most crucially of all they have devalued their core product and closed the window of opportunity for a word of mouth recommendation. If the Internet at large is to continue as a communal free space, it needs to look at a different model for raising revenue. People are catching on to the maxim "If it is free you are the product". While the majority are fine with tasteful, creative, discreet, secure and unobtrusive advertising (myself included), when the line is crossed into the territory of "You *will* watch our propaganda" I do what the majority of Britons do when an advert appears on commercial television. I go and make a cup of tea.

References

[1] *http://www.nature.com/news/injectable-brain-implant-spies-on-individual-neurons-1.17713*

ABOUT THE AUTHOR

Rob Somerville has been passionate about technology since his early teens. A keen advocate of open systems since the mid-eighties, he has worked in many corporate sectors including finance, automotive, airlines, government and media in a variety of roles from technical support, system administrator, developer, systems integrator and IT manager. He has moved on from CP/M and nixie tubes but keeps a soldering iron handy just in case.

#MISSIONCOMPLETE

“CryptoLocker is a joke with ZFS”

Learn how Plextec defeats ransomware attacks with FreeNAS and ZFS at iXsystems.com/cryptolocker

Have you used one of these tools to complete your mission?

Tell us more at iXsystems.com/missioncomplete for a chance to win monthly

#missioncomplete